Growth of graph states in quantum networks

Martí Cuquet and John Calsamiglia

Física Teòrica: Informació i Fenòmens Quàntics, Departament de Física, Universitat Autònoma de Barcelona,
E-08193 Bellaterra, Barcelona, Spain

We propose a scheme to distribute graph states over quantum networks in the presence of noise in the channels and in the operations. The protocol can be implemented efficiently for large graph sates of arbitrary (complex) topology. We benchmark our scheme with two protocols where each connected component is prepared in a node belonging to the component and subsequently distributed via quantum repeaters to the remaining connected nodes. We show that the fidelity of the generated graphs can be written as the partition function of a classical Ising-type Hamiltonian. We give exact expressions of the fidelity of the linear cluster and results for its decay rate in random graphs with arbitrary (uncorrelated) degree distributions.

PACS number(s): 03.67.Bg, 03.67.Hk, 89.75.Hc

I. INTRODUCTION

Quantum networks [1,2], where nodes with (limited) quantum storage and processing power are coupled through quantum channels, are becoming a focus of interest in quantum information. The first motivation is to extend the paradigmatic bipartite quantum communication applications (e.g., quantum teleportation [3] or more prominently quantum key distribution [4]) to a multipartite setting, where such bipartite protocols can be accomplished between arbitrary nodes of the network. The possibility of networks with nontrivial topologies can give rise to new phenomena and to applications that exploit the multipartite correlations. Of course, this puts forward a big variety of theoretical and technological challenges which can be addressed in short term. Indeed the first steps have already been taken towards a quantum network in a first realization with two distant nodes that can store and interchange quantum information in an efficient and reversible way [5]. Lastly, such physical realizations are in principle scalable and hence open the door to perform highly controllable experiments on many-body phenomena, study multipartite entanglement [6], and could eventually perform more complex tasks like some implementation of the quantum Google page rank [7] or general distributed quantum computations.

A central task in quantum networks is to devise strategies to distribute entanglement among its nodes. Linear networks have shown to be useful for long-distance bipartite entanglement distribution by means of quantum repeaters [8] (see also its measurement-based implementation [9]). However, the study of entanglement distribution over higher dimensional networks is in its infancy. Current results show that in some scenarios the network topology can bring interesting effects like entanglement percolation [10–16] that lead to new approaches to the problem. This effect shows up in nonideal (but still not fully realistic) scenarios or for some particular network geometries.

In this paper we address the problem of distributing graph states in the realistic scenario of noisy network channels and a small, but non-negligible, amount of noise in the local operations. Graph states [17] are a large family of multipartite entangled states that, although they can be efficiently described with relatively few parameters, have a rich variety of features. For instance, they include paradigmatic states like GHZ,

cluster states, codewords of error-correcting codes [18], they provide some novel quantum communication applications, like secret entangled-state distribution [19], they are useful in the study of nonlocality [20], and most importantly they include states which are universal resources for measurement-based quantum computing [21,22]. Current experiments [23,24] have succeeded in the experimental implementation of GHZ and cluster states up to 14 qubits [24].

Our goal here is to propose a protocol to create a large graph using an underlying network of noisy channels. The protocol should tolerate channels and operations with errors, scale efficiently with the size of the network, and work for any network topology, and in particular for complex networks. Complex networks [25–27] underlie many natural [28,29], social [30,31], and artificial systems [32,33] where different parties interact. Their nontrivial structure is the source of features, like the existence and shape of an important fraction of highly connected nodes [34] and the tendency of nodes to cluster together [30], that are often observed in real graphs and have a deep impact in their performance. Complex networks are of particular importance in communication infrastructures, as most present telecommunication networks like the Internet have a complex structure [32,33]. However, the rich and intriguing properties of these type of networks are still quite unexplored in the quantum setting (see nonetheless [11,13,15,35,36]).

Channels linking separate nodes in a network are typically noisy, and pose the main caveat to the creation of distributed multipartite entangled states with high fidelity. To overcome this, there exist bipartite [37,38] and multipartite [39–43] entanglement purification protocols that allow one either to generate highly purified Bell pairs, which can later be used to teleport an arbitrary graph state, or to directly purify the desired graph state (for a review, see Ref. [44]). These recursive protocols tolerate a reasonable amount of noise in local operations, but require a number of initial copies that grows exponentially with the size of the state. Other proposals do not use postselection, making the purification efficient in terms of the size of the graph state, but come at the expense of a stricter noise threshold [45]. In Ref. [46], an entanglement pumping without postselection is also used to obtain efficient purification when constructing the graph

state edge by edge. In different approaches, the graph state is created by a probabilistic growth using nondeterministic entangling operations [47–51].

Here we investigate the advantages of generating and purifying small GHZs that reproduce the local structure of the network, and merge them in order to distribute a network-wide graph state. Since GHZ have a fixed size which depends on the degree of each node and is thus independent of the size of the network, this protocol is efficient in the size while it still maintains the high thresholds of the recurrence schemes. One could also distribute different states, and create a large graph state with a structure different from that of the network recalling that, once the graph is created, local operations [52,53] might be used to transform the graph state and reach the desired structure. Note that different graph states might be locally equivalent under these types of transformations, but the differences in their structure can translate into a different sensibility to noise in their generation. We benchmark this protocol with two other protocols that generate high-fidelity bipartite states between a node and the rest of the network, which are then used to distribute a locally generated graph state. We use the fidelity of the graph state as a figure of merit to compare the three protocols. The fidelity decays exponentially with the size of the network for a constant level of noise, and hence we also use its decay rate. It turns out that both quantities can be understood as the partition function and free energy of a thermodynamic system, respectively, and thus standard methods of statistical mechanics are readily used.

The paper is structured as follows. First, in Sec. II we introduce graph states and the noise model we consider. In Sec. III we present the three protocols and the two figures of merit—the fidelity of the graph state and its decay rate—and relate them with an analog partition function and free energy of a thermodynamical system. Then, in Sec. IV we apply the protocols to the creation of a linear cluster state, for which we obtain exact results, and of a graph state associated with a complex network. We conclude in Sec. V.

II. DEFINITIONS

A. Graph states

A graph $G = \{V(G), E(G)\}$ is an ordered pair of sets: the set $V(G)$ of vertices, or nodes, and the set $E(G)$ of edges, or links, whose elements are (unordered) pairs of vertices and represent the connections between them. The neighborhood of a vertex $u \in V(G)$ is the set of vertices connected to it, $\mathcal{N}_u(G) = \{v : (u,v) \in E(G)\}$, and the degree of that vertex is the number of its neighbors, $k = |\mathcal{N}_u(G)|$. If the graph is *directed*, then the elements of $E(G)$ are ordered, and $(u,v) \in E(G)$ is a directed edge from u to v. In this case, the incoming and outgoing neighborhoods of u are $\mathcal{N}_u^{(\text{in})}(G) = \{v : (v,u) \in E(G)\}$ and $\mathcal{N}_u^{(\text{out})}(G) = \{v : (u,v) \in E(G)\}$, respectively.

A graph state [17] is a quantum state associated with a graph G, where vertices correspond to qubits and edges to interactions. A common description of graph states is the stabilizer formalism. For a graph G, there are $N = |V(G)|$ stabilizer operators (one for each vertex u) defined as

$$K_u^G = X_u \prod_{v \in \mathcal{N}_u(G)} Z_v. \tag{1}$$

Here X, Y, and Z are the Pauli operators and the subindex denotes on which qubit they operate. A pure graph state $|\boldsymbol{\mu}\rangle_G$, with $\boldsymbol{\mu} = \{0,1\}^N$, is a common eigenstate of all stabilizer operators with

$$K_u^G |\boldsymbol{\mu}\rangle_G = (-1)^{\mu_u} |\boldsymbol{\mu}\rangle_G \quad \forall \quad u \in V,$$

where μ_u is the uth component of $\boldsymbol{\mu}$. The set $\{|\boldsymbol{\mu}\rangle_G\}$ form the graph state basis. We will omit G when the graph is clear by context.

An alternative description is by means of the interaction picture, where $|\mathbf{0}\rangle_G$ is created by preparing all qubits in the state $|+\rangle$ and then applying a CPHASE$_{u,v}$ for every edge $(u,v) \in E$,

$$|\mathbf{0}\rangle_G = \prod_{(u,v) \in E} \text{CPHASE}_{u,v} |+\rangle^{\otimes N}.$$

Since Z_u anticommutes with K_u and commutes with the rest of K_v, $v \neq u$, any graph-basis element can be expressed as

$$|\boldsymbol{\mu}\rangle_G = \prod_u (Z_u)^{\mu_u} |\mathbf{0}\rangle_G.$$

Throughout this paper we consider mixed graph states diagonal in the graph state basis, $\rho_G = \sum_{\boldsymbol{\mu}} \lambda_{\boldsymbol{\mu}} |\boldsymbol{\mu}\rangle\langle\boldsymbol{\mu}|$. Any mixed graph state can be brought to this form by local depolarization [40]. We will work, however, in the stabilizer basis. Let us define $\boldsymbol{K}_{\boldsymbol{x}} = \prod_{u \in V} (K_v)^{x_v}$, where $\boldsymbol{x} = (x_1 x_2 \cdots x_N) \in \{0,1\}^N$. Since $\{\boldsymbol{K}_{\boldsymbol{x}}\}$ form a complete set of commuting observables, a diagonal graph state can be expressed as $\rho_G = \frac{1}{2^N} \sum_{\boldsymbol{x}} \langle \boldsymbol{K}_{\boldsymbol{x}} \rangle \boldsymbol{K}_{\boldsymbol{x}}$, where $\langle \boldsymbol{K}_{\boldsymbol{x}} \rangle = \sum_{\boldsymbol{\mu}} \lambda_{\boldsymbol{\mu}} (-1)^{\boldsymbol{\mu} \cdot \boldsymbol{x}}$. The effect of the Pauli operators on $\boldsymbol{K}_{\boldsymbol{x}}$ is

$$Z_u \boldsymbol{K}_{\boldsymbol{x}} Z_u = (-1)^{x_u} \boldsymbol{K}_{\boldsymbol{x}}, \tag{2}$$

$$X_u \boldsymbol{K}_{\boldsymbol{x}} X_u = \prod_{v \in \mathcal{N}_u} (-1)^{x_v} \boldsymbol{K}_{\boldsymbol{x}}, \tag{3}$$

$$Y_u \boldsymbol{K}_{\boldsymbol{x}} Y_u = (-1)^{x_u} \prod_{v \in \mathcal{N}_u} (-1)^{x_v} \boldsymbol{K}_{\boldsymbol{x}}, \tag{4}$$

so these unitaries map diagonal graph states into diagonal graph states. A CPHASE$_{u,v}$ adds an edge between u and v, if they were not connected, or removes it, if the edge already existed. The effect on $\boldsymbol{K}_{\boldsymbol{x}}$ is

$$\text{CPHASE}_{u,v} \boldsymbol{K}_{\boldsymbol{x}} \text{CPHASE}_{u,v} = \boldsymbol{K}_{\boldsymbol{x}} (Z_v)^{x_u} (Z_u)^{x_v}. \tag{5}$$

The action of Pauli measurements can also be easily described in this formalism as a transformation of the graph (up to some local unitaries). Measurement of Z simply disconnects the measured qubit from the rest of the graph, while X and Y transform the neighborhood of the measured qubit and then disconnect it. In terms of the stabilizer operators, the measurement of Z_u commutes with all K_v, for $v \neq u$, and anticommutes with K_u. Thus,

$$[I + (-1)^m Z_u] \boldsymbol{K}_{\boldsymbol{x}} [I + (-1)^m Z_u] = [I + (-1)^m Z_u] \boldsymbol{K}_{\boldsymbol{x}} \delta_{0,x_u}, \tag{6}$$

where $m = \{0,1\}$ labels the measurement outcome $\{+1,-1\}$, respectively. After tracing out qubit u, the new stabilizer is

$$(-1)^{m \cdot \sum_{v \in \mathcal{N}_u} x_v} \boldsymbol{K}'_{\boldsymbol{x}} \delta_{0,x_u}, \tag{7}$$

where the new $\boldsymbol{K}'_{\boldsymbol{x}}$ corresponds to a new graph G' obtained from G by removing vertex u and its attached edges. The

spurious phase factor in Eq. (7) can be canceled by applying a unitary $(\prod_{v \in \mathcal{N}_u} Z_v)^m$. Similarly, measurements of X_u or Y_u also result in the disconnection of the measured qubit, but in these cases the remaining graph is transformed by local complementations of the neighborhood of u, as described in Ref. [17]. We will be more explicit in the concrete cases where we apply these measurements.

Our figure of merit will be given in terms of the fidelity, which for a graph-diagonal state can be written as

$$F_N^G = {}_G\langle \mathbf{0}|\rho_G|\mathbf{0}\rangle_G = \lambda_{\mathbf{0}} = \frac{1}{2^N}\sum_{\mathbf{x}}\langle \mathbf{K}_{\mathbf{x}}^G\rangle. \quad (8)$$

B. Network and noise model

We consider a network where nodes are spatially separated. Some of the nodes are connected by a link, which we model as a noisy depolarizing channel on one qubit with error parameter p_c,

$$T_c^{(u)} = (1 - p_c)[I] + \frac{p_c}{4}\sum_{i=0}^{3}[\sigma_i^{(u)}]. \quad (9)$$

The square brackets $[A]$ denote that A acts on both sides of the state (i.e., $[A]\rho = A\rho A^\dagger$), and σ_i are I, X, Y, Z for $i = 0, 1, 2, 3$, respectively. A Pauli measurement on u is modeled as a perfect measurement preceded by a depolarizing channel $T_1^{(u)}$ with error probability p_1 on that qubit,

$$T_1^{(u)} = (1 - p_1)[I] + \frac{p_1}{4}\sum_{i=0}^{3}[\sigma_i^{(u)}]. \quad (10)$$

A noisy two-qubit gate (e.g., a CPHASE or a CNOT) on qubits u and v is modeled as an ideal gate followed by the two-qubit depolarizing channel on u, v with error parameter p_2,

$$T_2^{(u,v)} = (1 - p_2)[I] + \frac{p_2}{16}\sum_{i,j=0}^{3}[\sigma_i^{(u)}\otimes\sigma_j^{(v)}]. \quad (11)$$

Gates can only be applied locally (i.e., on qubits within the same node).

In the stabilizer basis, the effect of each noise source is easily tracked: It multiplies each stabilizer element by a coefficient, and keeps the graph state in diagonal form. Let us consider, for example, the effect of noise $T_2^{(u,v)}$ on $\mathbf{K}_{\mathbf{x}}$. The first part of Eq. (11), that is proportional to $1 - p_2$, does not affect $\mathbf{K}_{\mathbf{x}}$. Each term in the second part, that is proportional to p_2, will at most induce a change of sign. Using Eqs. (2)–(4) one easily finds that the sign change can only occur if $\mathbf{K}_{\mathbf{x}}$ acts nontrivially on u or v (i.e., if exists at least one $x_a = 1$ for $a \in u \cup v \cup \mathcal{N}_u \cup \mathcal{N}_v$). In addition one sees that half of the 16 terms in the sum induce a sign change which cancel the contribution the other half. Hence, the noise $T_2^{(u,v)}$ does not alter $\mathbf{K}_{\mathbf{x}}$ unless $\mathbf{x}$ has support in u or v, in which case it gets multiplied by a factor $(1 - p_2)$:

$$T_2^{(u,v)}(\mathbf{K}_{\mathbf{x}}) = (1 - p_2)^{\theta(x_u, x_v, \mathbf{x}_{\mathcal{N}_u}, \mathbf{x}_{\mathcal{N}_v})}\mathbf{K}_{\mathbf{x}}, \quad (12)$$

where $\theta(\mathbf{x}) \equiv 1 - \delta(\mathbf{0}, \mathbf{x})$ and $\mathbf{x}_{\mathcal{N}_u} = (x_{v_1}\cdots x_{v_k})$ for all $v_a \in \mathcal{N}_u$. Noise $T_1^{(u)}$ behaves similarly. In this case, the sign is

changed with probability $p_1/2$ unless $x_u = 0$ and $\bigoplus_a x_a = 0$ for $a \in \mathcal{N}_u$, so the multiplying coefficient is

$$(1 - p_1)^{\theta(x_u, \bigoplus_{v \in \mathcal{N}_u} x_v)}. \quad (13)$$

Note thus that these noise sources affect the qubits on which the gates act, plus their neighbors.

Finally let us point out that we do not associate any noise with the local "correcting" unitaries performed in order to bring the postmeasurement states to a standard graph form. We refer to these unitaries in our protocols to simplify bookkeeping, but their action can be pushed forward (or commuted) until the end of the protocol, and hence the resulting state is exactly equivalent as a resource of entanglement.

III. PROTOCOLS

We propose a protocol to distribute a graph state with the structure of a general quantum communication network, of arbitrary topology, associated with graph G. Noise p_c in the communication channels is considered to be relatively high, so some sort of purification or error correction is in order. The protocol uses the structure of the network to distribute several copies of small subgraphs between neighbors, and purifies them by means of multipartite purification. These subgraphs are GHZ states, which are associated with a star graph with a central node of degree j, connected to j leaves of degree 1. In order to benchmark our protocol, we also consider two reference protocols that distribute bipartite states between a central node, which locally creates the desired graph state, and the rest of the network. The bipartite states are then used to teleport the locally created graph state. Since this central node may not be directly connected to the rest of the network, quantum repeaters [8] are used to establish purified bipartite states between this node and all the network's nodes.

In all cases, we consider the same multipartite purification protocol for bicolorable graph states described in Ref. [40]. The noise threshold of this purification scheme depends on the degree of the central qubit of the state that is purified, and goes from ≈ 0.06 for a bipartite state (i.e., a GHZ with central degree 1) to ≈ 0.02 for a GHZ of central degree 9. This poses a limit in the maximum degree of a network for which the protocols can be used. Hence, we work under the assumption that the errors in the local operations are low—as compared to the (finite) degree of the subgraphs, $jp \ll 1$—and that the channel noise can be significant, but also low enough so that the minimum threshold fidelity required for the purification protocol to succeed can be attained. At first order in $p_1 = p_2 = p$, the output state fixed point of this purification scheme is given by (see Appendix A):

$$\left\langle K_a^{x_a}\prod_{b \in \mathcal{N}_a} K_b^{x_b}\right\rangle = 1 - p\left[\overline{x_a}\left\lceil\frac{\sum_b x_b}{2}\right\rceil + x_a(j+1)\right], \quad (14)$$

where a is the central node and b the leaves of the GHZ, j is the number of leaves and the overline in x_a represents the bit complement, $\overline{x_a} = x_a \oplus 1$.

Note that the extensive use of quantum repeaters in the first two protocols renders them extremely inefficient. Nevertheless, we will find that the performance in terms of attainable fidelity is still comparable to that of our more efficient subgraph protocol.

A. Bipartite A protocol

In the first protocol, purified, but still noisy, entangled states are created between a central node and all the others by means of quantum repeaters. This central node then teleports a locally generated state to all the other nodes. For simplicity, in protocols A and B we assume that the central node is sending *all* qubits through purified channels, including his own. This adds a source of noise that would not be strictly necessary, but its effect is small for large enough networks. In cases, like complex networks, where the network is not necessarily connected, it is understood that there is a central node for each connected component distributing the corresponding graph state.

The local state ρ_G, which mimics the structure of the network, is created by initializing N qubits in the state stabilized by X and then applying CPHASEs between neighbors. Then, the noisy Bell states between the central node and all the others are used to teleport the corresponding qubits (see Fig. 1). We label qubits in the local graph as $u \in V(G)$. For each node in the network, there is a bipartite state ρ_{g_u} of two qubits, $u_a, u_b \in V(g_u)$ and one edge $(u_a, u_b) \in E(g_u)$. Qubit u_a belongs to the central node, while u_b is in the corresponding node in the network. In order to account for the errors in the teleportation Bell measurement, we implement it by a CPHASE on (u, u_a) followed by X measurements on u and u_a.

Without taking noise into account, the state before teleportation is

$$\rho_G \otimes \bigotimes_{u \in V(G)} \rho_{g_u}, \tag{15}$$

with

$$\rho_G = \frac{1}{2^N} \sum_x K_x^G \quad \text{and} \quad \rho_{g_u} = \frac{1}{4} \sum_{x_{u_a}, x_{u_b}} \left(K_{u_a}^{g_u}\right)^{x_{u_a}} \left(K_{u_b}^{g_u}\right)^{x_{u_b}}.$$
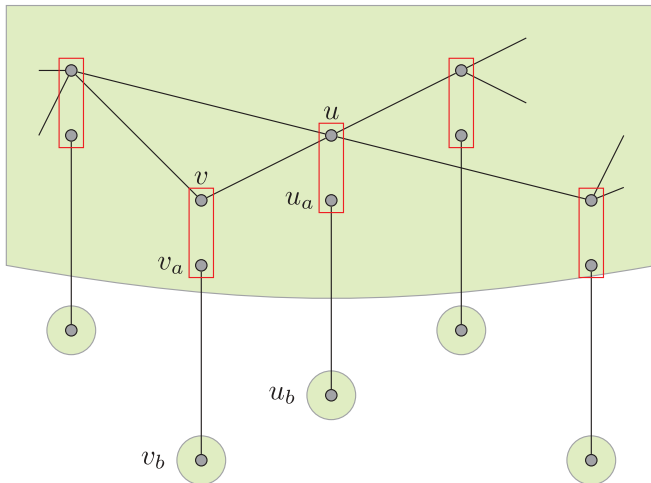

FIG. 1. (Color online) Bipartite A protocol. The upper green (gray) area represents the "central" node, where the local graph state with the structure of the network is generated. The lower green (gray) circles are nodes in the network. Small, dark gray dots correspond to qubits, and lines connect neighbors. Rectangles (in red) indicate Bell measurements involved in the teleportation of the local graph states, which are implemented as CPHASEs between u and u_a and X measurements at u, u_a.

The graphs g_u correspond to the bipartite states with qubits u_a and u_b used for teleportation (i.e., $K_{u_a}^{g_u} = X_{u_a} Z_{u_b}$). The action of the CPHASE$_{u,u_a}$ affects only the stabilizers $K_u^G \rightarrow K_u^G Z_{u_a}$ and $K_{u_a}^{g_u} \rightarrow K_{u_a}^{g_u} Z_u$. The measurement of X_u anticommutes with all K_v^G, $v \in \mathcal{N}_u(G)$, and with $K_{u_a}^{g_u} Z_u$, while that of X_{u_a} anticommutes only with $K_u^G Z_{u_a}$ and $K_{u_b}^{g_u}$. Thus, each term changes to

$$\left[1 + (-1)^{m_{u_a}} X_{u_a}\right]\left[1 + (-1)^{m_u} X_u\right] K_x,$$
$$\left(Z_{u_a}\right)^{x_u} \left(K_{u_a}^{g_u} Z_u\right)^{x_{u_a}} \left(K_{u_b}^{g_u}\right)^{x_{u_b}}, \tag{16}$$
$$\delta_{0, x_{u_a} \oplus \bigoplus_{v \in \mathcal{N}_u} x_v} \, \delta_{0, x_u \oplus x_{u_b}},$$

where m_u and m_{u_a} are the measurement outcomes. Tracing out qubits u and u_a and correcting the state with $(Z_{u_b})^{m_u}(X_{u_b})^{m_{u_a}}$ we obtain $K_x' \delta_{0, x_{u_a} \oplus \bigoplus_{v \in \mathcal{N}_u} x_v} \delta_{0, x_u \oplus x_{u_b}}$, with K_x' associated with a new graph G' where vertex u has been substituted by vertex u_b. Teleportation of all local qubits results in the desired distributed state.

Noise can now be introduced as the multiplicative factors in front of the stabilizer elements of Eq. (15). Here, the order of the CPHASE gates used to generate the local graph matters, as qubits receive noise from gates performed at neighbors which are already connected to them. There is thus a noise $(1 - p_1)^{x_u}$ corresponding to the preparation of each node in X_u, and a $(1 - p_2)^{\theta(x_u, x_v, x_{\mathcal{N}_u}, x_{\tilde{\mathcal{N}}_v})}$ for each edge in the local state, where the tilde in $\tilde{\mathcal{N}}_u$ labels that we consider the neighborhood of u at the moment the CPHASE$_{u,v}$ is performed. Additionally, and using that $x_{u_a} = \bigoplus_{v \in \mathcal{N}_u} x_v$ and $x_{u_b} = x_u$, in each teleportation the CPHASE introduces noise as $(1 - p_2)^{\theta(x_u, x_{\mathcal{N}_u})}$, and the measurements as $(1 - p_1)^{x_u + \bigoplus_{v \in \mathcal{N}_u} x_v}$. Finally, there is a $\langle (K_{u_a}^{g_u})^{\bigoplus_{v \in \mathcal{N}_u} x_v} (K_{u_b}^{g_u})^{x_u} \rangle$ term from the purified Bell states. This results in a final distributed state $\rho = \frac{1}{2^N} \sum_x \langle K_x \rangle K_x$ with

$$\langle K_x \rangle = \prod_{u \in V} \left[(1 - p_1)^{x_u} \langle (K_{u_a}^{g_u})^{\bigoplus_{v \in \mathcal{N}_u} x_v} (K_{u_b}^{g_u})^{x_u} \rangle \right.$$
$$\left. \times (1 - p_2)^{\theta(x_u, x_{\mathcal{N}_u})} (1 - p_1)^{x_u + \bigoplus_{v \in \mathcal{N}_u} x_v} \right]$$
$$\times \prod_{(u,v) \in E}^{\sim} (1 - p_2)^{\theta(x_u, x_v, x_{\tilde{\mathcal{N}}_u}, x_{\tilde{\mathcal{N}}_v})}, \tag{17}$$

where the tilde over Π states that the edges are introduced in a certain order. An explicit expression for the Bell state correlators for the postselection purification protocol at first order in p can be obtained from Eq. (14)

$$\left\langle \left(K_{u_a}\right)^{x_{u_a}} \left(K_{u_b}\right)^{x_{u_b}} \right\rangle \sim 1 - \left(\bar{x}_{u_a} x_{u_b} + 2 x_{u_a}\right) p. \tag{18}$$

B. Bipartite B protocol

The previous strategy can be improved by directly connecting the local ends of Bell pairs by means of CPHASEs, and then teleporting the local graph state performing only one X measurement per node—instead of the various sources of noise induced by the Bell measurement (CPHASE and two X measurements). The initial state is now $\bigotimes_{u \in V(G)} \rho_{g_u}$, where qubits are labeled as u if they belong to the central node and u_b if they are in the distributed nodes (see Fig. 2). The sources of noise are as follows: the purity of Bell pairs, CPHASEs used in the preparation of the local state, and the measurement involved in each teleportation. As in the previous case, the order of CPHASEs is important: Each contributes to a noise
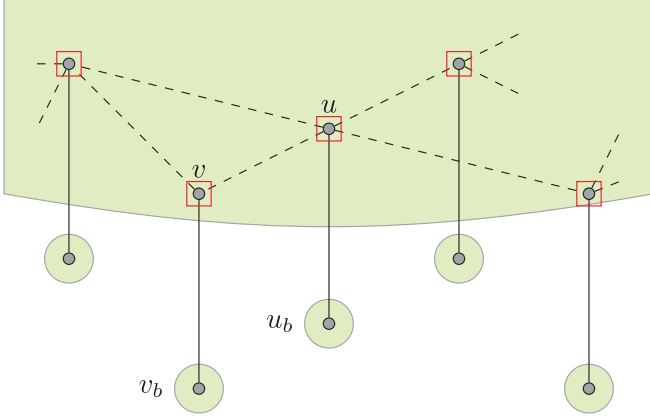
FIG. 2. (Color online) Bipartite B protocol. The upper green (gray) area represents the "central" node, where the local graph state with the structure of the network is generated. The lower green (gray) circles are nodes in the network. Small, dark gray dots correspond to qubits, and lines connect neighbors. Dashed lines, corresponding to the edges of the local graph, are created after solid lines, corresponding to the distributed bipartite states. Squares (in red) indicate X measurements used to teleport the local graph states.

with $(1 - p_2)^{\theta(x_u, x_v, x_{u_b}, x_{v_b}, \boldsymbol{x}_{\tilde{\mathcal{N}}_u}, \boldsymbol{x}_{\tilde{\mathcal{N}}_v})}$. Note that the CPHASES are performed on qubits that are already connected to nodes in the network and thus affect the distributed qubits u_b. The calculation is similar as before and results in a final distributed state with correlators,

$$
\langle \boldsymbol{K_x} \rangle = \prod_{u \in V} \left[\left\langle \left(K_u^{g_u} \right)^{x_u} \left(K_{u_b}^{g_u} \right)^{\bigoplus_{v \in \mathcal{N}_u} x_v} \right\rangle (1 - p_1)^{x_u} \right]
$$
$$
\times \widetilde{\prod_{(u,v) \in E}} (1 - p_2)^{\theta(x_u, x_v, \bigoplus_{w \in \mathcal{N}_u} x_w, \bigoplus_{w \in \mathcal{N}_v} x_w, \boldsymbol{x}_{\tilde{\mathcal{N}}_u}, \boldsymbol{x}_{\tilde{\mathcal{N}}_v})}.
$$
(19)

C. Purify subgraph and merge

In this strategy, subgraph states of small size (N independent) are distributed and purified, and then interconnected locally at each node to form the desired structure.

The protocol follows two steps (see Fig. 3). To each node of degree k we assign an outgoing neighborhood, with j nodes, and an incoming neighborhood, with i nodes ($k = j + i$). First, each node prepares a GHZ of size $j + 1$. A GHZ graph state with $j + 1$ qubits is associated with a star graph g_u (i.e., a graph with a central node and j external nodes, called leaves). Then, each qubit corresponding to a leaf of the GHZ is sent to one of the outgoing neighboring nodes through the depolarizing channels. Several copies of this distributed state are created and then purified using the bicolorable graph state protocol of Ref. [39]. The final purified subgraph state with central qubit u is

$$
\rho_u = \frac{1}{2^j} \sum_{x_u, \boldsymbol{x}_{\mathcal{N}_u^{(\text{out})}}} \left\langle \left(K_u^{g_u} \right)^{x_u} \prod_{u_w \in \mathcal{N}_u^{(\text{out})}} \left(K_{u_w}^{g_u} \right)^{x_{u_w}} \right\rangle
$$
$$
\times \left(K_u^{g_u} \right)^{x_u} \prod_{u_w \in \mathcal{N}_u^{(\text{out})}} \left(K_{u_w}^{g_u} \right)^{x_{u_w}},
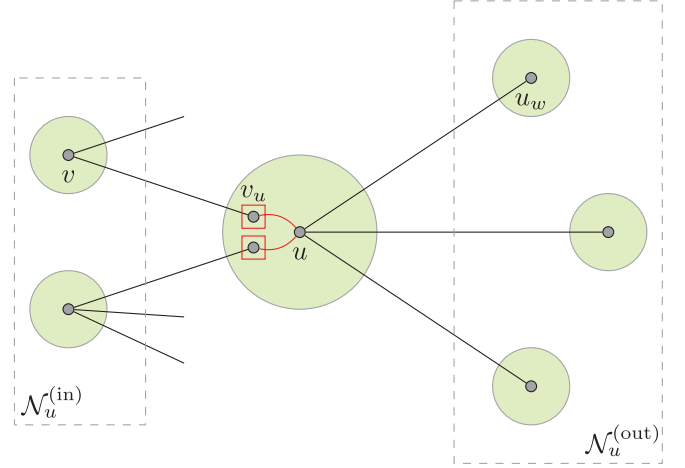$$
(20)

FIG. 3. (Color online) Subgraph purification and connection protocol. Green (gray) circles are nodes in the network. Nodes on the right are the outgoing neighborhood of u, while those on the left are the incoming neighborhood. Small, dark gray dots correspond to qubits, and lines connect neighbors. Lines and squares in red (gray, in the central node) indicate CPHASES and measurements involved in the connection of subgraphs.

where the correlators of the fixed point of the purification scheme are given in Eq. (14). Here u_w denotes the leaf qubit that has been sent to $w \in \mathcal{N}_u^{(\text{out})}(G)$. At the same time, each node receives $i = k - j$ leaves corresponding to the GHZ states created at its incoming neighborhood.

In the second step, each node connects the central qubits of their GHZ state with the leaves they have received. The connection is made by performing a CPHASE between the two qubits u and v_u and a Y measurement of the received qubit v_u. The effect of the Y measurement is to add an edge between the two central nodes u and v and destroy the measured qubit v_u.

The action on the stabilizers can be seen in a single connection example between a central qubit u of a subgraph and a leaf qubit v_u of an incoming subgraph (see Fig. 3). We focus on the stabilizer elements $(K_v^{g_v})^{x_v} (K_{v_u}^{g_v})^{x_{v_u}} (K_u^{g_u})^{x_u}$, as all the others remain unchanged. The action of a CPHASE between u and v_u is $K_{v_u}^{g_v} \rightarrow K_{v_u}^{g_v} Z_u$ and $K_u^{g_u} \rightarrow K_u^{g_u} Z_{v_u}$. The measurement of Y_{v_u} anticommutes with the new stabilizer operators at v, v_u, and u, so each term changes to

$$
\left[1 + (-1)^m Y_{v_u} \right] \left(K_v^{g_v} \right)^{x_v} \left(K_{v_u}^{g_v} Z_u \right)^{x_{v_u}} \left(K_u^{g_u} Z_{v_u} \right)^{x_u},
$$
$$
\delta_{0, x_v \oplus x_{v_u} \oplus x_u}.
$$
(21)

Tracing out v_u and correcting the state depending on the measurement outcome m with $\exp[(-1)^m i \frac{\pi}{4} Z_v] \exp[(-1)^m i \frac{\pi}{4} Z_u]$, we obtain

$$
(K_v')^{x_v} \left(K_u^{g_u} Z_v \right)^{x_u} \delta_{0, x_v \oplus x_{v_u} \oplus x_u},
$$

where K_v' is the stabilizer of v with qubit v_u changed to u.

Noise added by the measurement enters as $(1 - p_1)^{x_u \oplus x_v}$, where we used Eq. (13) together with $x_{v_u} = x_u \oplus x_v$. Noise by a CPHASE between one of the incoming leaves v_u and u adds $(1 - p_2)^{\theta(x_u, x_v, \boldsymbol{x}_{\mathcal{N}_u^{(\text{in})}}, \boldsymbol{x}_{\tilde{\mathcal{N}}_u^{(\text{in})}})}$, where $\tilde{\mathcal{N}}_u^{(\text{in})}$ is the incoming neighborhood that has already been connected to u.

Finally, note that if a node has $j = 0$ it does not need to prepare any GHZ state. In this case, one of the incoming leaves

is used as the qubit to which all the other leaves are connected. If there are no incoming leaves, then the node is isolated (and is thus prepared in the state stabilized by X). This means that the contribution of the GHZ distribution and purification at a node u is

$$\langle K_x^{g_u} \rangle = \begin{cases} \langle (K_u^{g_u})^{x_u} \prod_{w \in \mathcal{N}_u^{(\text{out})}} (K_{uw}^{g_u})^{x_u \oplus x_w} \rangle & \text{if } j > 0, \\ 1 & \text{if } j = 0, \ i > 0, \\ (1 - p_1)^{x_u} & \text{if } j = i = 0. \end{cases}$$

All in all, the final correlators for the subgraph protocol can be written as

$$\langle K_x \rangle = \prod_{u \in V} \left\{ \langle K_x^{g_u} \rangle \prod_{v_a \in \mathcal{N}_u'} \left[(1 - p_1)^{x_u \oplus x_{v_a}} \right. \right.$$

$$\left. \left. \times (1 - p_2)^{\theta(x_u, x_{\mathcal{N}_u^{\text{out}}}, x_{v_1}, \dots, x_{v_a})} \right] \right\}. \quad (22)$$

Here, $\mathcal{N}_u'$ denotes the incoming neighborhood $\mathcal{N}_u^{\text{in}}$ if $j > 0$, or the incoming neighborhood minus the first incoming neighbor, $\mathcal{N}_u^{\text{in}} \backslash v_1$, in the case $j = 0$ where the first neighbor v_1 is used as the qubit to which all remaining edges are connected.

D. Fidelity

The fidelity for a distributed graph G is $F_N^G = \frac{1}{2^N} \sum_x \langle K_x^G \rangle$, and is in general hard to calculate. In order to simplify the analysis here we will use a single error parameter $p = p_1 = p_2$ —most of the results presented here can be extended to more general dependencies between the CPHASE and measurement errors, which strongly depend on the physical implementation. With this particular parametrization the effect of errors enter as factors,

$$(1 - p)^{h(x)} = e^{-\beta h(x)}, \quad (23)$$

where we have defined $\beta = -\ln(1 - p)$. To first order in p, the correlators of the purified graph states (14) can also be written in the same form,

$$1 - h(x)p \sim e^{-p h(x)} \sim e^{-\beta h(x)}. \quad (24)$$

Hence, the correlators of the generated graph state can be written as $\langle K_x \rangle = \exp[-\beta H^G(x)]$, where $H^G(x)$ is the sum of the different noise terms $h(x)$. The fidelity thus resembles the partition function of a system with Hamiltonian $H^G(x)$ and inverse temperature β:

$$F_N^G = \frac{1}{2^N} \sum_x e^{-\beta H^G(x)}. \quad (25)$$

The "Hamiltonian" $H^G(x)$ can be expressed as the sum of many-body n-local Hamiltonians of the form,

$$\theta(x_1, x_2, \dots, x_n) = 1 - \overline{x_1} \, \overline{x_2} \cdots \overline{x_n}, \quad (26)$$

$$x_u \oplus x_v = x_u \overline{x_v} + \overline{x_u} x_v, \quad (27)$$

$$\overline{x_u} \left\lceil \frac{\sum_{b=1}^j x_b}{2} \right\rceil = \frac{1}{2} \left(\overline{x_u} \sum_{b=1}^j x_b + \overline{x_u} \bigoplus_{b=1}^j x_b \right). \quad (28)$$

The last term $\bigoplus_{b=1}^j x_b$ in Eq. (28) is a j-body interaction term. By recursively using Eq. (27), this term can be seen to be equal to the sum over all index permutations of

$\sum_a$ odd $x_1 \cdots x_a \overline{x_{a+1}} \cdots \overline{x_j}$, that is, we have rephrased our problem of computing the fidelity of a distributed large graph state as that of computing the thermal properties of a classical many-body Ising-type system, where the indices x take the role of classic spins. The corresponding Hamiltonian will inherit the topology of the underlying graph and its precise expression will depend on the graph-growth protocol used.

We are interested in the rate at which F_N^G decays,

$$\beta f_N^G = -\frac{1}{N} \ln F_N^G, \quad (29)$$

where $N f_N^G$ is the analog of the free energy of the system. A good reason to study this quantity is that in statistical systems such as a complex network (which is modeled as an ensemble of graphs, each associated with a probability—see Sec. IV B), the partition function (fidelity) in itself is not an extensive quantity, while the free energy is typically extensive and self-averaging (see, e.g., Ref. [54], p. 188).

We can further exploit the statistical physics analogy and apply the known methods and understanding to compute the rate at which the fidelity decays for the different proposed protocols. We are interested in a regime where the noise in the operations is low, which corresponds to the high temperature limit. In addition, in the cases under study, each spin (or node) is effectively coupled to several spins, either as nearest- or second-nearest neighbors. These are conditions for which mean-field approximation is very well suited: Eqs. (26)–(28) can be linearized using the standard mean-field approximation to express the Hamiltonian as the sum of a constant term plus linear terms in x_u. For this purpose we take $x_u \to s + \delta_u$ (and $\overline{x_u} \to 1 - s - \delta_u$) where s is the value of the mean field and δ_u are the arguably small fluctuations of x_u around its mean value. Keeping only the linear terms in the fluctuations, the different terms present in the Hamiltonian become

$$\theta(x_1, x_2, \dots, x_n) = 1 - (1 - s)^n + (1 - s)^{n-1} \sum_{a=1}^n \delta_a, \quad (30)$$

$$x_u \oplus x_v = 2s(1 - s) + (1 - 2s)(\delta_u + \delta_v), \quad (31)$$

$$\overline{x_u} \left\lceil \frac{\sum_b x_b}{2} \right\rceil$$
$$= \frac{1 - s}{2} \left(js + \sum_{b=1}^j \delta_b \right) - \frac{\delta_u}{2} js + \frac{1 - s}{2}$$
$$\times \left\{ \frac{1}{2} [1 - (1 - 2s)^j] + (1 - 2s)^{j-1} \sum_{b=1}^j \delta_b \right\}$$
$$- \frac{\delta_u}{2} \frac{1}{2} [1 - (1 - 2s)^j]. \quad (32)$$

With this linearization, the new mean-field Hamiltonian takes the form $H^{\text{MF}} = \sum_u A_u + \sum_u B_u x_u$, where A_u and B_u are functions of s. Hence, the sum over x (i.e., "configurations") in the fidelity can now be carried out with ease,

$$F = \frac{1}{2^N} \sum_x \prod_{u \in V} e^{-\beta A_u} e^{-\beta B x_u}$$

$$= \prod_{u \in V} e^{-\beta A_u} e^{-\beta B/2} \left(\cosh \frac{\beta B_u}{2} \right), \quad (33)$$

and the decay rate becomes βf^{MF}, with

$$f^{\text{MF}} = \frac{1}{N} \sum_u A_u + \frac{1}{N} \sum_u \frac{B_u}{2} - \frac{1}{\beta N} \sum_u \ln \cosh \frac{\beta B_u}{2}.$$

(34)

The value of s is found by adding an artificial linear term to the Hamiltonian (playing the role of an external magnetic field), changing $B_u \to B_u + \xi$, and requiring consistency in the definition,

$$s = \langle x_u \rangle = \left. \frac{\partial f^{\text{MF}}}{\partial \xi} \right|_{\xi \to 0}.$$

One is hence left with the trascendental equation for s,

$$s = \frac{1}{2} - \frac{1}{2N} \sum_u \tanh \frac{\beta B_u}{2}.$$

(35)

Its solution s^* can be substituted back in the expression for f^{MF} (34) to obtain the desired result. In the cases of interest here, B_u is some polynomial which remains bounded for all values of $s \in [0,1]$. Hence, to leading order in p we can approximate $\tanh \frac{\beta B_u}{2} \approx \frac{p B_u}{2}$, arriving at

$$s \approx \tfrac{1}{2} - \tfrac{1}{2} \, p B_u|_{s=1/2},$$

(36)

for small enough p.

IV. NETWORK EXAMPLES

A. Closed linear cluster

We first study the case of a one-dimensional network, in which a linear cluster state is created. This case is remarkable because we can compute the exact fidelity for any cluster size. For symmetry, we consider a closed linear cluster state, where all nodes have degree 2. The order of the CPHASEs in the creation of the local graph in bipartite A and B protocols, and the size of subgraphs in protocol subgraphs, gives different results for the fidelity. For simplicity, in bipartite A and B, we consider CPHASEs applied to successive nodes. Except for the noise of the first gate (which affects only two nodes) and of the last one (which affects four nodes), all the gates contribute to the noise of three nodes. Each of the correlators is thus

$$\langle \boldsymbol{K_x} \rangle \simeq \prod_{u \in V} (1-p_1)^{x_u} (1-p_1)^{x_u + x_{u-1} \oplus x_{u+1}}$$
$$\times (1-p_2)^{2\theta(x_{u-1},x_u,x_{u+1})} \big\langle \big(K_{u_a}^{g_u}\big)^{x_{u-1} \oplus x_{u+1}} \big(K_{u_b}^{g_u}\big)^{x_u} \big\rangle$$

(37)

for the bipartite A protocol and

$$\langle \boldsymbol{K_x} \rangle \simeq \prod_{u \in V} (1-p_1)^{x_u} (1-p_2)^{\theta(x_{u-1},x_u,x_{u+1},x_{u+2})}$$
$$\times \big\langle \big(K_u^{g_u}\big)^{x_u} \big(K_{u_b}^{g_u}\big)^{x_{u-1} \oplus x_{u+1}} \big\rangle$$

(38)

for the bipartite B protocol. Recall that the correlator of the purified subgraphs is given in Eq. (14).

The subgraph protocol, on the other hand, depends on the size of the subgraphs. There are two extreme strategies, depicted in Fig. 4: create subgraphs of degree 1, and connect them successively (S1), or create subgraphs of degree 2 every
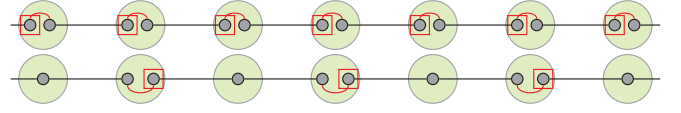

FIG. 4. (Color online) Subgraph protocol in a linear cluster. (Top) S1, using subgraphs of degree 1. (Bottom) S2, using subgraphs of degree 2.

second node, and connect them at their common neighbor (S2). One could also adopt an intermediate strategy, in which the degree of each GHZ is selected at random, resulting in a mix of subgraphs of degrees 1 and 2 —its performance, which will not be reported here, falls in between the two extreme strategies. In the first case (S1), Eq. (22) reads

$$\langle \boldsymbol{K_x} \rangle = \prod_{u \in V} (1-p_2)^{\theta(x_u,x_{u+1},x_{u+2})} (1-p_1)^{x_u \oplus x_{u+1}}$$
$$\times \big\langle \big(K_u^{g_u}\big)^{x_u} \big(K_{u+1}^{g_u}\big)^{x_u \oplus x_{u+1}} \big\rangle,$$

(39)

while in the second (S2) it is

$$\langle \boldsymbol{K_x} \rangle = \prod_{u \in V_{\text{even}}} (1-p_2)^{\theta(x_u,x_{u+1},x_{u+2})} (1-p_1)^{x_{u+1} \oplus x_{u+2}}$$
$$\times \big\langle \big(K_u^{g_u}\big)^{x_u} \big(K_{u_{u-1}}^{g_u}\big)^{x_u \oplus x_{u-1}} \big(K_{u_{u+1}}^{g_u}\big)^{x_{u+1}} \big\rangle.$$

(40)

Note that in the latter the product is over $u = 2,4,\cdots \in V_{\text{even}}$.

Let us now define a domain of $\boldsymbol{x} = (x_1 x_2 \cdots x_N)$ as a sequence $(x_u x_{u+1} \cdots x_{u+l})$ where all x_v, for $v = u,\ldots,u+l$, have the same value (either 0 or 1), and where x_{u-1} and x_{u+l+1} have a different value. We can differentiate between a sequence of zeros or ones. Taking Eq. (39) as an example, one can observe that except for the term $(1-p_2)$, $\langle \boldsymbol{K_x} \rangle$ factorizes into expected values in each component of ones, as the exponents of the noise terms can be different from zero if at least one of the bits x is equal to one, and the terms are uncorrelated with those on different domains. Considering all noise terms, Eq. (39) can be expressed as a function of the total number of ones in the sequence $\boldsymbol{x}$, $n = |\boldsymbol{x}|$; the number c_1 of domains of only one zero; and the number of domains c_{2*} with two or more zeros:

$$\langle \boldsymbol{K_x} \rangle = (1-p_2)^{c_1+2c_{2*}+n} (1-p_1)^{2(c_1+c_{2*})}$$
$$\times \big\langle K_{u_v}^{g_u} \big\rangle^{c_1+c_{2*}} \big\langle K_u^{g_u} \big\rangle^{n-c_1-c_{2*}} \big\langle K_u^{g_u} K_{u_v}^{g_u} \big\rangle^{c_1+c_{2*}}. \quad (41)$$

The fidelity is then,

$$F_N = \frac{1}{2^N} \sum_{c_1,c_{2*},n} g(c_1,c_{2*},n,N) \langle \boldsymbol{K_x}(c_1,c_{2*},n) \rangle,$$

where $g(c_1,c_{2*},n,N)$ is the number of $\langle \boldsymbol{K_x}(c_1,c_{2*},n) \rangle$ elements with these parameters in a graph of N vertices. This sum can be computed exactly by turning to generating functions. The function generating $g(c_1,c_{2*},n,N)$ is $G(x,y_1,y_2,z) = \sum g(c_1,c_{2*},n,N)x^n y_1^{c_1} y_2^{c_{2*}} z^N$. We derive its closed form in Appendix B, which reads

$$G = \frac{1 - xz^2(1 - y_1 + 2(y_1 - y_2)z)}{1 - (1+x)z + x(1-y_1)z^2 + x(y_1 - y_2)z^3}.$$

(42)

The fidelity for a state of size N is then

$$F = \frac{1}{2^N} \frac{1}{N!} \partial_z^N G(x,y_1,y_2,z),$$
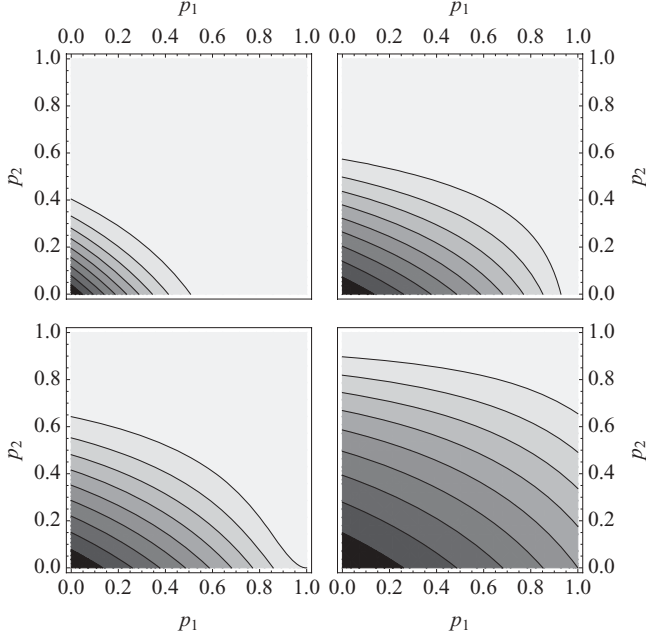
(43)

FIG. 5. Fidelity of a linear cluster of size $N = 10$ created by bipartite A (top left) and B (top right), and subgraph S1 (bottom left) and S2 (bottom right) protocols, as a function of noise parameters p_1 and p_2. Lines correspond to fidelities $2^{-1}, 2^{-2}, \ldots, 2^{-9}$, starting from the bottom left corner.

evaluated at

$$x = (1 - p_2)\langle K_u^{g_u}\rangle, \tag{44}$$

$$y_1 = (1 - p_2)(1 - p_1)^2 \langle K_{u_v}^{g_u}\rangle \langle K_u^{g_u}\rangle^{-1} \langle K_u^{g_u} K_{u_v}^{g_u}\rangle, \tag{45}$$

$$y_2 = (1 - p_2)^2(1 - p_1)^2 \langle K_{u_v}^{g_u}\rangle \langle K_u^{g_u}\rangle^{-1} \langle K_u^{g_u} K_{u_v}^{g_u}\rangle, \tag{46}$$

and $z = 0$. The factorization of $\langle \boldsymbol{K_x}\rangle$ for subgraph S2 and bipartite protocols depend on slightly different parameters, and the corresponding generating functions can also be computed (see Appendix B).

We will first consider the ideal case of perfect purification, and later tackle the noisy case. In the ideal scenario, we can substitute all subgraph correlators by 1 and Eq. (43) gives the exact fidelity for all values of p_1 and p_2 (here taken to be independent parameters). The fidelities of the four protocols are plotted in Fig. 5, the best protocol being S2. The fidelity decays at first exponentially in the error probabilities but as probability of error increases the decay becomes slower and the fidelity reaches a nonzero minimum value. In general, the fidelity is more sensible to noise p_2, as this is the one corresponding to CPHASEs, which affects more qubits. The dependence on p_1 varies a lot from one protocol to another. This is reflected in the asymptotic minimum value: For $p_2 = 0$ and $p_1 = 1$ (i.e., perfect two-qubit gates but maximally noisy single-qubit measurements), the fidelity in the S1 and S2 protocols is higher than the minimum value 2^{-N}. This happens when the exponent of $(1 - p_1)^{f(\boldsymbol{x})}$ in Eqs. (39) and (40) is zero (i.e., the corresponding $\boldsymbol{K_x}$ is not affected by the noise). Of course, this is true in all protocols for the identity, $\boldsymbol{x} = \boldsymbol{0}$, but in the S1 it is also true for $\boldsymbol{x} = \boldsymbol{1}$, so the minimum fidelity is 2^{-N+1}. In protocol S2 the effect is even bigger, as the product

is over V_{even} and hence any $\boldsymbol{x}$ where pairs $x_{2n+1} \oplus x_{2n+2} = 0$ for $n = 0, \ldots, N/2 - 1$ translates in a $\boldsymbol{K_x}$ that is not affected by this noise. In this case, the minimum fidelity is $2^{-N/2}$.

To further compare the four protocols, in Fig. 6 we plot the corresponding decay exponent f_N for $p = p_1 = p_2$. As expected, the behavior of the bipartite A protocol is the most sensible to noise, because it involves more operations and measurements. One can also observe that S1 and bipartite B have a similar decay rate. Note that, except for the use of a central node in bipartite B, while S1 protocol is distributed, both use bipartite resources, and in particular both implement the same number of CPHASEs and measurements. For this reason, noise comes in a very similar way, the main difference being that CPHASEs in bipartite B affect a larger number of qubits, and therefore the associated θ function has a larger support in $\boldsymbol{x}$. This holds even if one considers nonperfect purification, as both protocols rely on bipartite purification. On the other hand, S2 performs the merging of subgraphs at every second node, so it has much fewer sources of noise. In fact, for $p \to 0$ its decay rate is exactly half the one of S1.

We now compare these exact results with the mean-field approximation. As we introduced in the previous section, $\langle \boldsymbol{K_x}\rangle \sim \exp[-\beta H(\boldsymbol{x})]$. For the subgraph S1 protocol, $H(\boldsymbol{x}) = \sum_{u \in V} h_u(\boldsymbol{x})$, with

$$h_u(\boldsymbol{x}) = \theta(x_u, x_{u+1}, x_{u+2}) + x_u \oplus x_{u+1}. \tag{47}$$

Recall that we are still considering perfect purification. Substituting $x_u \to s + \delta_u$ and keeping only linear terms in δ_u, this takes the form of $h_u^{\text{MF}}(\boldsymbol{x}) = a_u + b_u x_u + c_u x_{u+1} + d_u x_{u+2}$, with

$$a_u = (5 - 2s)s^2, \tag{48}$$

$$b_u = 2 - 4s + s^2, \tag{49}$$

$$c_u = 2 - 4s + s^2, \tag{50}$$
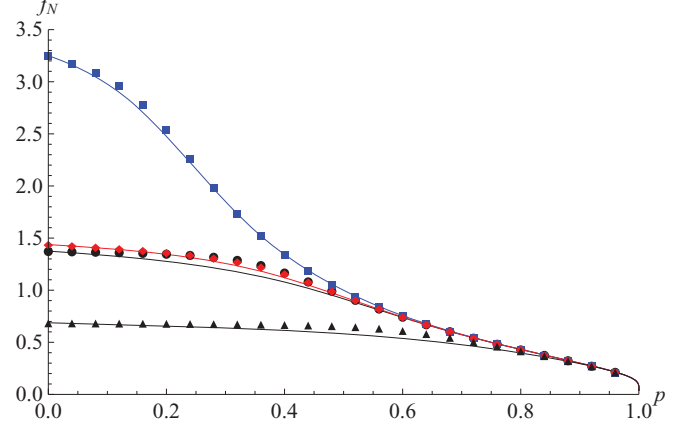
$$d_u = (1 - s)^2. \tag{51}$$

FIG. 6. (Color online) Rescaled decay rate f_N of a closed linear cluster of $N = 10$ nodes with perfect purification. (From upper to lower lines) Bipartite A (blue, square), bipartite B (red, diamonds), subgraph S1 (black, circles), and subgraph S2 (black, triangles). Dots correspond to the mean-field approximation, lines to the exact result using generating functions.

All qubits u are equivalent, so the coefficients a,b,c,d are the same for every node and the total Hamiltonian is $H^{\mathrm{MF}} = AN + B \sum_u x_u$, with $A = a$ and $B = b + c + d$. Hence, the fidelity in the mean-field approximation is

$$F_N^{\mathrm{MF}} = e^{-\beta AN} e^{-\beta BN/2} \left(\cosh \frac{\beta B}{2} \right)^N, \qquad (52)$$

and the decay rate,

$$f^{\mathrm{MF}} = A + \frac{B}{2} - \frac{1}{\beta} \ln \cosh \frac{\beta B}{2}. \qquad (53)$$

The same approximation can be made for the other protocols. The local Hamiltonian of subgraph S2 is

$$h_u(\boldsymbol{x}) = \theta(x_u, x_{u+1}, x_{u+2}) + x_{u+1} \oplus x_{u+2}. \qquad (54)$$

Note that, in this case, the total Hamiltonian is the sum for u even. In the case of bipartite A and B, the local Hamiltonians are

$$h_u(\boldsymbol{x}) = 2x_u + x_{u-1} \oplus x_{u+1} + 2\theta(x_{u-1}, x_u, x_{u+1}), \quad (55)$$

and

$$h_u(\boldsymbol{x}) = x_u + \theta(x_{u-1}, x_u, x_{u+1}, x_{u+2}), \qquad (56)$$

respectively. The mean-field results are shown in Fig. 6 together with the exact result. The agreement is remarkably good, especially considering that this is a one-dimensional network configuration. We observe that the decay rate has a linear dependence in p but soon higher order (nonlinear) terms start to kick in. Here, we have studied and plotted the solution for a wide range of p. This has been done for completeness and to check the validity of the mean-field approximation, but recall that the aim in this paper is solely to compute the decay rate to leading order in p. The reason being that in realistic scenarios p will be strongly limited by threshold values required for subgraph purification.

We now consider the realistic scenario following the same procedure as before but taking into account the corrections due to the noisy purified subgraphs. This can be done by approximating the correlators by their linear correction around unity (14). Consequently, an additional term (24) must be added to the local Hamiltonians (47), (54), (55), and (56) for each of the different protocols. It is important to bear in mind that all subsequent results have to be taken consistently up to leading order in p. This leading order is the one that enters in the features that we seek: the linear dependence of the fidelity decay rate around $p \sim 0$ (for arbitrarily large N). From Eqs. (34) and (35) it is immediate to see that at this order the whole contribution to the decay rate is given by the first two terms in (34) evaluated at $s = 1/2$, that is,

$$f_N = A + \frac{B}{2} \bigg|_{s=1/2} = \frac{1}{N2^N} \sum_{\boldsymbol{x}} H(\boldsymbol{x}). \qquad (57)$$

The second equality is straightforward to get from the definitions of A and B, and it states that in this regime (low p) the decay rate is dominated by the exponents of the typical sequences $\boldsymbol{x}$. The expected value of $H(\boldsymbol{x})$ (over sequences $\boldsymbol{x}$) can be easily carried out since x_u are independent variables of mean $1/2$ and it leads to fidelities that decay as $p f_N N$ with f_N
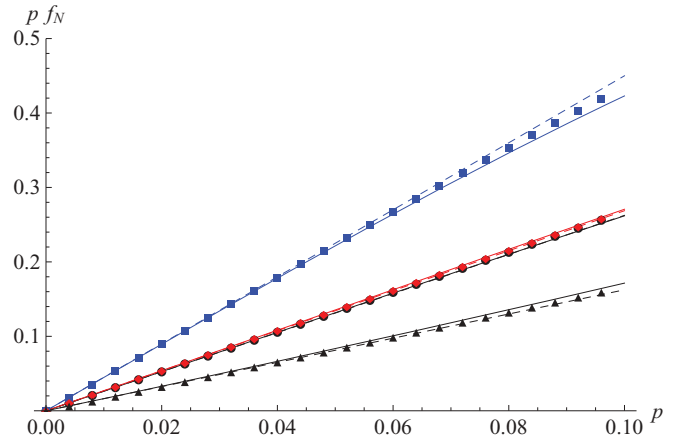

FIG. 7. (Color online) Fidelity decay rate $p f_N$ of a closed linear cluster of $N = 100$ nodes. (From upper to lower lines) Bipartite A (blue, squares), bipartite B (red, circles), subgraph S1 (black, diamonds), and subgraph S2 (black, triangles). Solid lines correspond to the generating function result, using the first-order approximation of the purification scheme of Eq. (14). Dashed lines are the first-order result of Eq. (57). Dots correspond to the mean-field approximation.

equal to $9/2 = 4.5$, $43/16 \approx 2.7$, $21/8 \approx 2.6$, and $13/8 \approx 1.6$ for bipartite A and B and subgraphs S1 and S2, respectively.

These coincide with the decay rates obtained by the generating function method in the same limit, where in Eqs. (44)–(46) we substitute $\langle K_u^{g_u} \rangle = \langle K_u^{g_u} K_{u_v}^{g_u} \rangle = 1 - 2p$ and $\langle K_{u_v}^{g_u} \rangle^{c_1+c_2*} = 1 - p$. The results (see Fig. 7) show that the subgraph protocols provide output fidelities comparable, if not better, than those given by the protocols based on channel purification. This is remarkable keeping in mind that purifying channels is much more demanding in terms of resources and efficiency.

As we mentioned at the beginning of the section, the order of the CPHASEs in the bipartite protocols can give different results, but the decay rates do not change much. For example, if the CHPASEs are first applied to every second edge, and then to the remaining edges, the decay rate is $143/32 \approx 4.5$ for bipartite A, while it does not change for bipartite B.

B. Complex networks

Now we study the behavior of the protocols in complex networks. Complex networks are characterized by statistical properties, and can be modeled as an ensemble of graphs $\mathcal{G}$, with a probability $P(G)$ assigned to every graph G in the ensemble. A property O of a complex network is defined as its average over the ensemble, $\overline{O} = \sum_{G \in \mathcal{G}} P(G) O^G$ (note that the overline here stands for the ensemble average). Some of these properties are *self-averaging*, meaning that for large systems (in the limit $N \to \infty$) a property of a given graph realization G is the same as the average over different realizations of $G \in \mathcal{G}$ [54,55]. As we discussed in Sec. III D, the free energy is typically self-averaging, and is the one we use here as a figure of merit.

We consider random networks with uncorrelated degree distribution p_k, which defines the ensemble. This family of networks includes the Erdős-Rényi model (when p_k is a Poisson distribution) as well as exponential and scale-free

networks [34]. It is useful to express the degree distribution by its generating function $g_p(z) = \sum_k p_k z^k$. Related to p_k is the probability that, following a random edge, we arrive at a vertex with k other edges. This probability is $r_k = (k + 1)p_{k+1}/\langle k \rangle$, and k is called the *excess degree*. It is generated by $g_r(x) = g'_p(x)/\langle k \rangle$. The edges in the network are undirected, but the creation of the graph state via the subgraph protocol is "directed-like," as each node can have incoming and outgoing neighbors. Thus, we also consider the degree distribution $p_{i,j}$, where i is the in-degree and j the out-degree. This distribution defines the implementation of the protocol. For example, in the linear cluster, S1 was defined by $p_{1,1} = 1$ for all nodes, and S2 by $p_{0,2} = 1$ and $p_{2,0} = 1$ at even and odd nodes, respectively. The distribution $p_{i,j}$ is constrained by $\sum_{i,j}(j - i)p_{i,j} = 0$, which means that $\langle i \rangle = \langle j \rangle = \langle k \rangle/2$, but note that, in general, it can be correlated ($p_{i,j} \neq p_i p_j$). The function that generates $p_{i,j}$ is $g_p(x, y) = \sum_{i,j} p_{i,j} x^i y^j$.

In the case of complex networks, we are interested in the average $\overline{f_N} = \sum_{G \in \mathcal{G}} P(G) f_N^G$ over the ensemble. We expect that, for large N, f_N goes to $\overline{f}$ due to self-averaging. In order to average f_N we have to compute

$$\overline{f_N} = -\frac{1}{\beta N} \sum_{G \in \mathcal{G}} P(G) \ln \frac{1}{2^N} \sum_{\boldsymbol{x}} e^{-\beta H^G(\boldsymbol{x})},$$

which corresponds to a quenched average, where the disorder corresponding to the network topology, given by $P(G)$, is frozen with respect to that of the correlation operator index $\boldsymbol{x}$. Computing this quantity is in general an extremely challenging problem, especially in our case that we have structured disorder and k-body interaction between the spins. Nevertheless, we are interested in the particular regime of high temperatures (low p) far away from critical phenomena, long-range correlations, and other difficulties that appear at low temperatures. In this regime we can directly use (57)

$$\overline{f} = \frac{1}{N} \sum_{G \in \mathcal{G}} P(G) \frac{1}{2^N} \sum_{\boldsymbol{x}} H^G(\boldsymbol{x}) = \frac{1}{N} \overline{\frac{1}{2^N} \sum_{\boldsymbol{x}} H^G(\boldsymbol{x})}. \quad (58)$$

The Hamiltonians for the bipartite protocols are of the form,

$$H^G(\boldsymbol{x}) = \sum_{u \in V} h_u(\boldsymbol{x}) + \sum_{(u,v) \in E} \widetilde{h_{u,v}(\boldsymbol{x})}. \quad (59)$$

In bipartite A, the local Hamiltonians are

$$h_u(\boldsymbol{x}) = x_u + x_u \overline{\bigoplus_{v \in \mathcal{N}_u} x_v} + 2 \bigoplus_{v \in \mathcal{N}_u} x_v$$
$$+ \theta(x_u, \boldsymbol{x}_{\mathcal{N}_u}) + x_u + \bigoplus_{v \in \mathcal{N}_u} x_v, \quad (60)$$

and

$$h_{u,v}(\boldsymbol{x}) = \theta(x_u, x_v, x_{\tilde{\mathcal{N}_u'}}, x_{\tilde{\mathcal{N}_v'}}), \quad (61)$$

while in bipartite B,

$$h_u(\boldsymbol{x}) = \overline{x_u} \bigoplus_{v \in \mathcal{N}_u} x_v + 2x_u + x_u, \quad (62)$$

and

$$h_{u,v}(\boldsymbol{x}) = \theta\left(x_u, x_v, \bigoplus_{w \in \mathcal{N}_u} x_w, \bigoplus_{w \in \mathcal{N}_v} x_w, x_{\tilde{\mathcal{N}_u}}, x_{\tilde{\mathcal{N}_v}}\right). \quad (63)$$

As before, the tilde over the summatory and the neighborhood stands for the order in which CPHASEs are performed in the local graph. We can substitute it by the expected effect of an edge, which depends on the number of edges already connected to nodes u and v. This number is $n_u + n_v$ with probability $\frac{1}{2^{k_u+k_v}} \binom{k_u}{n_u} \binom{k_v}{n_v}$. Here, k_u and k_v are the excess degrees of the vertices in edge (u, v), so the average of the network ensemble has to be performed using probabilities r_{k_u} and r_{k_v}:

$$\overline{h_{u,v}(\boldsymbol{x})} = \frac{1}{2^N} \sum_{\boldsymbol{x}} \sum_{k_u k_v} r_{k_u} r_{k_v} h_{u,v}(\boldsymbol{x}). \quad (64)$$

In bipartite A, the average effect of each of these edges is $1 - \frac{1}{4}[g_r(3/4)]^2$. In bipartite B, it is $1 - \frac{1}{16}[g_r(1/2) + g_r(3/4)]^2$. The terms in $h_u(\boldsymbol{x})$, on the other hand, depend directly on the degree, and the average is performed over p_k:

$$\overline{h_u(\boldsymbol{x})} = \frac{1}{2^N} \sum_{\boldsymbol{x}} \sum_k p_k h_u(\boldsymbol{x}). \quad (65)$$

Considering all these terms, and that the summation over V contains N elements, while that over E contains $\langle k \rangle N/2$, the decay rates are

$$\overline{f_N} = \frac{15}{4} - \frac{5}{4} g_p(0) - \frac{1}{2} g_p(1/2) + \frac{\langle k \rangle}{2}\left(1 - \frac{1}{4}[g_r(3/4)]^2\right), \quad (66)$$

for bipartite A and

$$\overline{f_N} = \frac{7}{4} - \frac{1}{4} g_p(0) + \frac{\langle k \rangle}{2}\left(1 - \frac{1}{16}[g_r(3/4) + g_r(1/4)]^2\right), \quad (67)$$

for bipartite B.

In the subgraph protocol, where $H^G(\boldsymbol{x}) = \sum_{u \in V} h_u(\boldsymbol{x})$, the local Hamiltonian is

$$h_u(\boldsymbol{x}) = \left[\overline{x_u}\left\lceil\frac{\sum_{w \in \mathcal{N}_u^{\text{out}}} x_w}{2}\right\rceil + x_u(j^{(u)} + 1)\right](1 - \delta_{0,j^{(u)}})$$
$$+ x_u \delta_{0,i^{(u)}} \delta_{0,j^{(u)}} + \sum_{v_a \in \mathcal{N}_u'}\left[\theta(x_u, \boldsymbol{x}_{\mathcal{N}_u^{\text{out}}}, x_{v_1}, \ldots, x_{v_a})\right.$$
$$\left. + x_u \oplus x_{v_a}\right]. \quad (68)$$

Here $\mathcal{N}_u'$ is defined as in Eq. (22). In this case, to average the Hamiltonian we have to take into account the directed degree probability p_{ij}, $\overline{h(\boldsymbol{x})} = \sum_{ij} p_{ij} h(\boldsymbol{x})$, giving

$$\overline{f_N} = \frac{5}{8} + \frac{17\langle k \rangle}{16} + \frac{7}{4} g_p(0,0) - \frac{15}{8} g_p(1,0)$$
$$- \frac{1}{2}[g_p(1,1/2) - g_p(1/2,1/2)]. \quad (69)$$

These results are valid for any network with uncorrelated degree distribution p_k. To compare the behavior of the three protocols we consider the Erdős-Rényi model [56–58], which is the maximally random graph under the only constraint that the mean degree $\langle k \rangle$ is fixed. The degrees in this model
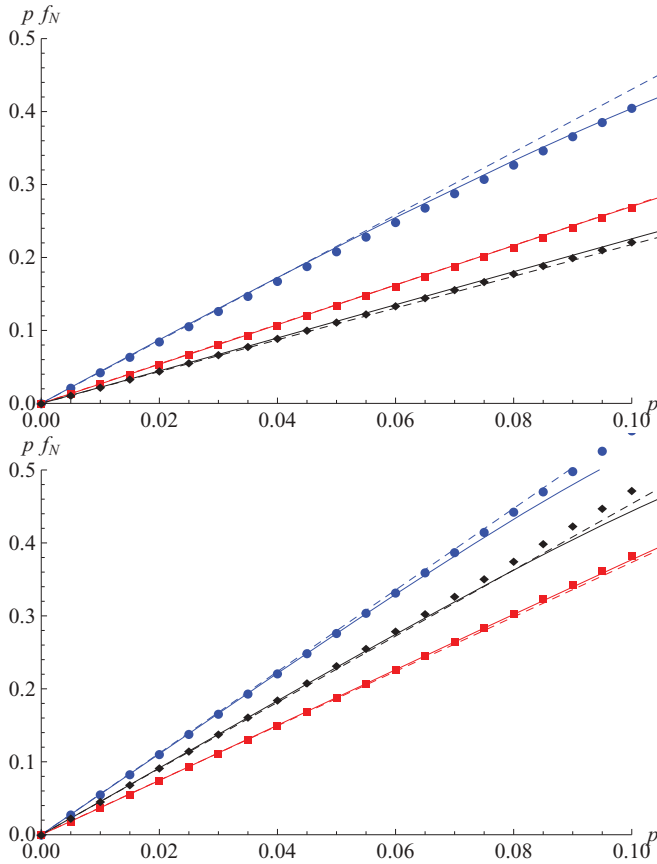
FIG. 8. (Color online) Fidelity decay rate $p\overline{f_N}$ in an Erdős-Rényi network of mean degree $\langle k \rangle = 2$ (top) and 4 (bottom). Bipartite A (blue, circles), bipartite B (red, squares), and subgraphs (black, diamonds). Bipartite A (upper, blue), subgraphs (middle, black), and bipartite B (lower, red). Dots are the average over 10 random realizations of networks with $N = 100$, randomly sampling 1000 configurations of $\boldsymbol{x}$. Dashed lines are the results at first order in p [Eqs. (66)–(68)].

follow a Poisson distribution, so $g_p(x) = g_r(x) = e^{\langle k \rangle (x-1)}$, and for $\langle k \rangle > 1$ there is a giant connected component of size comparable to the size of the network. The degree distribution p_k is fixed by the network model, but in the subgraph protocol one can play with the distribution $p_{i,j}$ (as long as $\sum_{i,j} p_{i,j} \delta_{i+j,k} = p_k$). For simplicity, here we consider that the direction of each edge is selected at random,

$$p_{i,j} = \frac{p_{i+j}}{2^{i+j}} \binom{i+j}{i}, \tag{70}$$

and hence $g_p(x,y) = e^{\langle k \rangle (\frac{x+y}{2}-1)}$. Figure 8 plots $p\overline{f_N}$ for the three protocols, showing that the best protocol depends on the mean degree of the network. It also shows an average over 10 random realizations of an Erdős-Rényi graph of $N = 100$, approximating the fidelity by the average of a random sample of 1000 configurations of $\boldsymbol{x}$, $F \sim \frac{1}{1000} \sum_{\boldsymbol{x}_{\text{sample}}} \langle \boldsymbol{K}_{\boldsymbol{x}_{\text{sample}}} \rangle$. This approximation is valid in the low p regime, where the fidelity is dominated by the typical values of $\boldsymbol{x}$. We observe that, in the Erdős-Rényi, the subgraph protocol with random edge direction give a better (lower) decay rate $\overline{f_N}$ for $\langle k \rangle < 2.8$, while above that bipartite B beats it. This stronger dependence

of the subgraph protocol on the mean degree of the network—compare the term in $\langle k \rangle$ in Eq. (68) versus those on Eqs. (66) and (67), which however have a bigger constant term—is in part due to the higher number of noise sources, which depends on the number of edges. Indeed, in the bipartite protocols only the initial CPHASEs depend on the number of edges of the graph, while in the subgraph protocol every merging of two subgraphs includes a CPHASE and a measurement. These mergings depend on the number of incoming edges, which could be lowered by considering a strategy different from the simpler one considered here where the direction of edges is selected at random. Instead, one could favor, for example, that the leaves of the network have an incoming edge with higher probability than an outgoing edge. Here it is important that we are comparing protocols which have drastically different requirements in terms of resources, so a slight benefit of the bipartite protocols in terms of fidelity does not rule out the use of subgraph protocols.

V. CONCLUSIONS

We have proposed a protocol to generate a graph state that spans a distributed network of any topology, in the presence of noise in the communication channels and in the local quantum operations performed in the network nodes. The protocol distributes and purifies small subgraphs, that are then merged to mimic the network structure. This allows the use of multipartite purification for small systems (that depend on the degree of the nodes, rather than on the size of the network), which makes the protocol scale efficiently. We have compared this protocol, which uses multipartite state purification and graph merging, to two other protocols that rely on channel purification (quantum repeaters).

We have benchmarked the protocols using as a figure of merit the fidelity of the generated graph state, or, more precisely, its decay rate, as the size of the network increases. Using generating function methods we have been able to compute exactly the fidelity for linear clusters of arbitrary size, allowing for a direct comparison with the approximate methods we develop. We have rephrased the problem of computing the effects of noise in the operations in terms of the thermal properties of a classical spin system, with the same interaction patterns as the underlying graph. Indeed, the fidelity itself can be seen as the analog of the partition function of such a system, while its decay rate plays the role of the free energy. The well-known methods from statistical physics, such as the mean-field approximation, can be used to study its behavior.

We have also studied the three protocols in networks with a complex structure. These complex networks are present in many real-world systems, and are of particular importance in the description of communication networks. The subgraph protocol is especially motivated to be implementable in networks of this kind, as it is highly adaptable to any network topology and it only requires the nodes to have local knowledge of the network.

Our results shows that the protocol using subgraph purification and merging is comparable, and in some cases even better, than those which rely on bipartite states. This is quite remarkable, as the latter rely on quantum repeaters and require

much more resources. In the complex networks case, the multipartite protocol could be possibly enhanced by devising an optimized strategy via the directed degree distribution $p_{i,j}$. As an example, in networks with many nodes of degree 1, one should go for a directed degree distribution in which those nodes are as much as possible the receivers of GHZ states, instead of the senders, so less connections would need to be made. In other words, one should tend to $p_{1,0}$ as high as possible (compared to $p_{0,1}$). In addition, in nodes of too high degree, the protocol might fail due to the noise threshold in the multipartite purification, which depends on the size of the GHZs (and thus, on the degree of the network nodes). In this case, one could always separate the node in two (or more) and treat them as independent nodes, each creating and distributing a subgraph among a subset of the neighborhood. This might prove useful in networks with a scale-free degree distribution, which have a long tail and a high presence of hubs.

It remains an open question to relate the decay in fidelity with the actual use one can make of graph states. Clearly, there is a regime where the fidelity is exponentially small, but a finite decay rate still signatures valuable resources. It is important to emphasize that for the three protocols, as well as for all the networks under study, we have not only computed this decay rate, but also completely characterized the noise graph states that emerge from the protocol. For a particular application, having full knowledge of the generated graph gives probably a closer picture of the generated resource.

APPENDIX A: MULTIPARTITE PURIFICATION PROTOCOL

The recursive protocol [40] is an entanglement purification protocol for multipartite, two-colorable graph states, which has also been extended to general graph states [41]. It operates on two identical copies,

$$\rho_1 \otimes \rho_2 = \frac{1}{2^{2N}} \sum_{\boldsymbol{x}_1, \boldsymbol{x}_2} \langle \boldsymbol{K}_{\boldsymbol{x}_1}^{(1)} \rangle \langle \boldsymbol{K}_{\boldsymbol{x}_2}^{(2)} \rangle \boldsymbol{K}_{\boldsymbol{x}_1}^{(1)} \boldsymbol{K}_{\boldsymbol{x}_2}^{(2)},$$

of a two-colorable graph state of size N (with colors A and B), and consists of two subprotocols (P1 and P2), each of which purifies one of the two colors. Here $\boldsymbol{K}_{\boldsymbol{x}_i}^{(i)}$ is a stabilizer element of state ρ_i. In each subprotocol, information about ρ_1 is transferred to ρ_2. Then, ρ_1 is kept or discarded depending on the outcomes of measurements on ρ_2. In P1, a CNOT$_{2\to1}$ is applied to every node in A, and a CNOT$_{1\to2}$ to every node in B. This transforms the state to

$$\frac{1}{2^{2N}} \sum_{\boldsymbol{x}_1, \boldsymbol{x}_2} \langle \boldsymbol{K}_{\boldsymbol{x}_1}^{(1)} \rangle \langle \boldsymbol{K}_{\boldsymbol{x}_2}^{(2)} \rangle \boldsymbol{K}_{\boldsymbol{a}_1 \oplus \boldsymbol{a}_2}^{(A_1)} \boldsymbol{K}_{\boldsymbol{b}_1}^{(B_1)} \boldsymbol{K}_{\boldsymbol{a}_2}^{(A_2)} \boldsymbol{K}_{\boldsymbol{b}_2 \oplus \boldsymbol{b}_1}^{(B_2)},$$

where $\boldsymbol{a}$ and $\boldsymbol{b}$ are the elements of $\boldsymbol{x}$ that correspond to colors A and B. The stabilizer $K_{\boldsymbol{a}_1 \oplus \boldsymbol{a}_2}^{(A_1)}$ corresponds to qubits of

ρ_1 in color A, and the modulo 2 summation of the index $\boldsymbol{a}_1 \oplus \boldsymbol{a}_2$ is made elementwise. To keep notation short, we contract $\boldsymbol{K}_{\boldsymbol{a}_1 \oplus \boldsymbol{a}_2}^{(A_1)} \boldsymbol{K}_{\boldsymbol{b}_2}^{(B_1)} = \boldsymbol{K}_{\boldsymbol{a}_1 \oplus \boldsymbol{a}_2, \boldsymbol{b}_2}^{(1)}$, where the first subindex corresponds to color A and the second to color B. Then, every node in ρ_2 is measured: X in nodes of color A (outcomes ξ) and Z in nodes of color B (outcomes ζ). This gives the (unnormalized) state,

$$\sum_{\boldsymbol{x}_1, \boldsymbol{x}_2} (-1)^{(\xi \oplus \bigoplus \zeta) \cdot \boldsymbol{a}_2} \langle \boldsymbol{K}_{\boldsymbol{x}_1}^{(1)} \rangle \langle \boldsymbol{K}_{\boldsymbol{x}_2}^{(2)} \rangle \boldsymbol{K}_{\boldsymbol{a}_1 \oplus \boldsymbol{a}_2, \boldsymbol{b}_1}^{(1)} \delta_{\boldsymbol{0}, \boldsymbol{b}_2 \oplus \boldsymbol{b}_1}$$

$$= \sum_{\boldsymbol{x}_1, \boldsymbol{a}_2} (-1)^{(\xi \oplus \bigoplus \zeta) \cdot \boldsymbol{a}_2} \langle \boldsymbol{K}_{\boldsymbol{a}_1, \boldsymbol{b}_1}^{(1)} \rangle \langle \boldsymbol{K}_{\boldsymbol{a}_2, \boldsymbol{b}_1}^{(2)} \rangle \boldsymbol{K}_{\boldsymbol{a}_1 \oplus \boldsymbol{a}_2, \boldsymbol{b}_1}^{(1)},$$

where $\xi \oplus \bigoplus \zeta$ stands for $\xi_u \oplus \bigoplus_{v \in \mathcal{N}_u} \zeta_v$ for all $u \in A$. The state is selected if $\xi \oplus \bigoplus \zeta = 0$. Summing over all the possible outcomes, the final (postselected) state after P1 is

$$\frac{1}{2^{N|A|}} \sum_{\boldsymbol{x}_1, \boldsymbol{a}_2} \langle \boldsymbol{K}_{\boldsymbol{a}_1}^{(1)} \boldsymbol{K}_{\boldsymbol{b}_1}^{(1)} \rangle_1 \langle \boldsymbol{K}_{\boldsymbol{a}_2}^{(2)} \boldsymbol{K}_{\boldsymbol{b}_1}^{(2)} \rangle_2 \boldsymbol{K}_{\boldsymbol{a}_1 \oplus \boldsymbol{a}_2}^{(A_1)} \boldsymbol{K}_{\boldsymbol{b}_1}^{(B_1)}.$$

P2 is equivalent, with colors A and B interchanged.

For our protocol, we need the fixed point for a GHZ of size $j + 1$, with a central node colored as A and j leaves colored as B ($|A| = 1$ and $|B| = j$). Noise can come from CNOTs as $1 - p_2$ and from measurements in state ρ_2 as $1 - p_1$:

$$\langle K_a \boldsymbol{K}_{\boldsymbol{b}} \rangle^{(P1)} = \frac{1}{2} \sum_{a_2=0}^{1} \langle K_{a \oplus a_2}^{(1)} \boldsymbol{K}_{\boldsymbol{b}}^{(1)} \rangle \langle K_{a_2}^{(2)} \boldsymbol{K}_{\boldsymbol{b}}^{(2)} \rangle$$

$$\times (1 - p_2)^{\theta(a, a_2, \bigoplus_{b \in B} b)} \prod_{b \in B} (1 - p_2)^{\theta(a, a_2, b)}$$

$$\times (1 - p_1)^{a_2} \prod_{b \in B} (1 - p_1)^{a_2},$$

$$\langle K_a \boldsymbol{K}_{\boldsymbol{b}} \rangle^{(P2)} = \frac{1}{2^d} \sum_{\boldsymbol{b}_2=\boldsymbol{0}}^{\boldsymbol{1}} \langle K_a^{(1)} \boldsymbol{K}_{\boldsymbol{b} \oplus \boldsymbol{b}_2}^{(1)} \rangle \langle K_a^{(2)} \boldsymbol{K}_{\boldsymbol{b}_2}^{(2)} \rangle$$

$$\times (1 - p_2)^{\theta(a, \bigoplus_{b \in B} b_2, \bigoplus_{b \in B} b)} \prod_{b \in B} (1 - p_2)^{\theta(a, \boldsymbol{b}_2, b)}$$

$$\times (1 - p_1)^{\bigoplus_{b \in B} b_2} \prod_{b \in B} (1 - p_1)^{b_2}.$$

We now consider $p_1 = p_2 = p$ and approximate $\langle K_a \boldsymbol{K}_{\boldsymbol{b}} \rangle$ at first order in p. Let the unnormalized $\langle K_a \boldsymbol{K}_{\boldsymbol{b}} \rangle^{(P1)} \sim 1 - \tilde{\beta}_{a, |\boldsymbol{b}|} p$ and $\langle K_a \boldsymbol{K}_{\boldsymbol{b}} \rangle^{(P2)} \sim 1 - \tilde{\alpha}_{a, |\boldsymbol{b}|} p$. Composing P1 and P2 we can find the fixed point at first order in p. In P1, each $\tilde{\beta}_{a, |\boldsymbol{b}|}$ equals $\tilde{\alpha}_{0, |\boldsymbol{b}|} + \tilde{\alpha}_{1, |\boldsymbol{b}|}$ plus a constant term:

$$\tilde{\beta}_{0, |\boldsymbol{b}|} = \alpha_{0, |\boldsymbol{b}|} + \alpha_{1, |\boldsymbol{b}|} + j + 1 + \left\lceil \frac{|\boldsymbol{b}|}{2} \right\rceil,$$

$$\tilde{\beta}_{1, |\boldsymbol{b}|} = \alpha_{0, |\boldsymbol{b}|} + \alpha_{1, |\boldsymbol{b}|} + \frac{3}{2}(j + 1).$$

These terms are normalized dividing them by $\langle \mathbb{1} \rangle = \langle K_0 \boldsymbol{K}_{\boldsymbol{0}} \rangle$, so the normalized first-order coefficients for P1 read $\beta_{a, |\boldsymbol{b}|} = \tilde{\beta}_{a, |\boldsymbol{b}|} - \tilde{\beta}_{0, 0}$. Similarly, for P2,

$$\tilde{\alpha}_{0, |\boldsymbol{b}|} = \frac{1}{2^j} \sum_{\boldsymbol{b}_2} \left(\beta_{0, \boldsymbol{b} \oplus \boldsymbol{b}_2} + \beta_{0, \boldsymbol{b}_2} \right) + j + 1 + \left\lceil \frac{|\boldsymbol{b}|}{2} \right\rceil,$$

$$\tilde{\alpha}_{1, |\boldsymbol{b}|} = \frac{1}{2^j} \sum_{\boldsymbol{b}_2} \left(\beta_{1, \boldsymbol{b} \oplus \boldsymbol{b}_2} + \beta_{1, \boldsymbol{b}_2} \right) + \frac{3}{2}(j + 1).$$

After normalization (dividing by $\langle K_0 K_0 \rangle$),

$$\langle K_0 K_b \rangle \sim 1 - \left\lceil \frac{|b|}{2} \right\rceil, \quad \langle K_1 K_b \rangle \sim 1 - (j+1)p.$$

The fidelity is

$$F \sim 1 - \frac{1}{2^{j+1}} \sum_{b=0}^{j} \binom{j}{b} \left[\left\lceil \frac{b}{2} \right\rceil + (j+1) \right] p$$

$$= 1 - \frac{5}{8}(j+1)p.$$

APPENDIX B: GENERATING FUNCTIONS FOR THE DOMAINS IN THE CLOSED LINEAR CLUSTER STATE

In this appendix we derive the generating functions from which the fidelity of a closed linear cluster can be derived using Eq. (43). In all cases, errors from CPHASES, measurements, and multipartite purification are tracked by error parameters p_2, p_1, and p, respectively.

1. Subgraph S1: $g(n, c_1, c_{2*}, N)$

In a closed linear chain, nodes can have values "0" or "1". In this chain, a domain is a sequence of adjacent nodes with the same value (0 or 1), surrounded by nodes of different value. In a chain of N nodes, let n be the total number of ones, c_1 the number of domains of ones preceded by a domain of only a zero, and c_{2*} the number of domains of ones preceded by a domain of at least two zeros($c_1 + c_{2*}$ is the total number of domains of ones). Recall Eq. (41), which gives the correlator of the protocol subgraph S1 in terms of n, c_1, and c_{2*}:

$$\langle K_x \rangle = (1 - p_2)^{c_1 + 2c_{2*} + n}(1 - p_1)^{2(c_1 + c_{2*})}$$
$$\times \langle K_{u_v}^{g_u} \rangle^{c_1 + c_{2*}} \langle K_u^{g_u} \rangle^{n - c_1 - c_{2*}} \langle K_u^{g_u} K_{u_v}^{g_u} \rangle^{c_1 + c_{2*}}.$$

Then, $g(n, c_1, c_{2*}, N)$ is the number of different configurations of that chain with given parameters, and

$$G(x, y_1, y_2, z) = \sum g(n, c_1, c_2, N) x^n y_1^{c_1} y_2^{c_{2*}} z^N$$

its generating function. Each variable x, y_1, y_2, and z "counts" the number of ones, domains of one zero, domains of two or more zeros, and the total size, respectively. The function G can be found by joining simpler distributions. We can think of the linear chain as a construction of domains of zeros and ones joined together. Consider the set of domains of zeros, $\{0, 00, 000, \dots\}$, each domain of a given size appearing only once. The number of domains of size N in this set is $d_N = 1$, which is generated by

$$D(z) = \sum_{n \geq 1} d_N z^N = \frac{z}{1 - z}.$$

We can differentiate between the set of domains of only a zero, $\{0\}$, and that of two or more zeros, $\{00, 000, \dots\}$. In this case, the generating functions are, respectively, z and $D(z) - z$. The set of domains of ones, $\{1, 11, 111, \dots\}$, is generated by the same function, but here each element contributes to the total size of the chain and to the number of ones. Its generating function is thus $D(xz)$. The function generating the set of pairs of domains, the first of zeros and the second of ones, is

$$P \equiv P(x, y_1, y_2, z) = \{y_1 z + y_2 [D(z) - z]\} D(xz).$$

The function for all possible combinations of pairs of domains of zeros and ones (in order) is

$$\frac{1}{1 - P} = 1 + \{y_1 z + y_2 [D(z) - z]\} \frac{D(xz)}{1 - P}.$$

Here, the first element (1) counts the case where there are no pairs at all, and the second, $P/(1 - P)$, to that where there is at least one pair. We can now add nothing at all, a possible domain of zeros at the end, a possible domain of ones at the beginning, or both. The final generating function is then

$$G = \left\{ 1 + [y_1 z + y_2 (D(z) - z)] \frac{D(xz)}{1 - P} \right\} [1 + D(xz) + P]$$
$$+ \left[1 + y_2 D(z) D(xz) \frac{1}{1 - P} \right] D(z).$$

Note that when we added a domain of zeros [the term with $D(z)$], we changed the variable y_1 for y_2, to take into account that the domain of zeros is now of size greater than one (because we are considering a closed linear chain). Simplifying, we obtain

$$G = \frac{1 - xz^2[1 - y_1 + 2(y_1 - y_2)z]}{1 - (1 + x)z + x(1 - y_1)z^2 + x(y_1 - y_2)z^3}. \quad \text{(B1)}$$

2. Bipartite B: $g(n, c_1, c_2, c_{3*}, N)$

In the bipartite B protocol, the correlator depends on similar parameters, but here we have to differentiate between the number of domains of zeros with one element (c_1), two elements (c_2), and three or more elements (c_{3*}):

$$\langle K_x \rangle = (1 - p_1)^n (1 - p_2)^{c_1 + 2c_2 + 3c_{3*} + n}$$
$$\times (1 - 2p)^n (1 - p)^{2c_2 + 2c_{3*}}. \quad \text{(B2)}$$

This can be achieved by a small modification of the previous generating function. Now,

$$P \equiv P(x, y_1, y_2, y_3, z)$$
$$= \{y_1 z + y_2 z^2 + y_3 [D(z) - z - z^2]\} D(xz),$$

and

$$G = \frac{\{1 + xz^2[-1 + y_1 - 2y_1 z + 2y_2 z + 3(-y_2 + y_3)z^2]\}}{\{1 + z[-1 + x(-1 + z\{1 + y_1(-1 + z) + z[y_2(-1 + z) - y_3 z]\})]\}}. \quad \text{(B3)}$$

3. Bipartite A: $g(n, c_1, c_{2^*}, \bar{c}_{2^*}, N)$

In bipartite A, we also need to count the number $\bar{c}_{2^*}$ of domains of two or more ones. The correlator in this case reads

$$\langle \boldsymbol{K_x} \rangle = (1 - p_1)^{2n + 2c_{2^*} + 2\bar{c}_{2^*}}(1 - p_2)^{2c_1 + 4c_{2^*} + 2n}$$
$$\times (1 - 2p)^{2c_{2^*} + 2\bar{c}_{2^*}}(1 - p)^{n - 2\bar{c}_{2^*}}. \qquad (B4)$$

We count $\bar{c}_{2^*}$ using variable w_2, and differentiating between the sets $\{1\}$ and $\{11, 111, \dots\}$, which are generated by xz and $D(xz) - xz$, respectively. The extended function generating the set of pairs of domains is now

$$P \equiv P(x, y_1, y_2, w_2, z)$$
$$= \{y_1 z + y_2 [D(z) - z]\}, \quad \{xz + w_2 [D(xz) - xz]\}.$$

Proceeding as in the previous case, we obtain

$$G = \frac{1 + xz^2(-1 + y_1 - 2y_1 z + 2y_2 z) + (-1 + w_2)x^2 z^3 [y_1(2 - 3z) + 3y_2 z]}{1 + z[-1 + x(-1 + z\{1 + [1 + (-1 + w_2)xz][y_1(-1 + z) - y_2 z]\})]}. \qquad (B5)$$

4. Subgraphs S2: $g(n_{01}, n_{10}, n_{11}, c_l, c_r, N)$

In the subgraph S2 protocol, the sum is performed over even nodes. In this case, it is convenient to express $\boldsymbol{x}$ as a sequence of elements 00, 01, 10, and 11 (the first digit corresponding to an odd node, and the second to an even node). Each 01, 10, and 11 contribute to one CPHASE noise, as well as each domain of 00 which is preceded by a 01 or a 11. Moreover, each 01 and 10 contribute to one Y measurement noise. Finally, each 01 and 11 contribute to a $1 - 3p$ noise of the purified subgraph, each 10 to a $1 - p$ and each domain of 00 followed by a 10 or an 11 also to a $1 - p$. Thus, we need the number of configurations with n_{01}, n_{10}, and n_{11} number of 01, 10, and 11 elements, and c_l and c_r domains of 00 preceded by 01 or 11 and followed by 10 or 11, respectively. The correlator is

$$\langle \boldsymbol{K_x} \rangle = (1 - p_2)^{n_{01} + n_{10} + n_{11} + c_l}(1 - p_1)^{n_{01} + n_{10}}$$
$$\times (1 - 3p)^{n_{01} + n_{11}}(1 - p)^{n_{10} + c_r}. \qquad (B6)$$

Now, each element contributes with z^2 to the size of the chain. The domains $\{00, 00\ 00, \dots\}$ are generated by $\frac{z^2}{1 - z^2}$. A domain made of elements 01, 10, and 11 of any size (including

0) is generated by

$$\frac{1}{1 - (x_{01} + x_{10} + x_{11})z^2},$$

and one which ends (or begins) with, say, element 01 (and thus is of size at least 2) is generated by

$$\frac{1}{1 - (x_{01} + x_{10} + x_{11})z^2}x_{01}z^2.$$

Proceeding as in the previous cases, the generating function is

$$G = \left[1 + \frac{(x_{01} + x_{10} + x_{11})z^2}{1 - (x_{01} + x_{10} + x_{11})z^2} + \frac{z^2}{1 - z^2} + G \right] \frac{1}{1 - P}, \qquad (B7)$$

where P is

$$\left\{ (x_{01}y_l + x_{10}y_r + x_{11}y_l y_r)z^2 + \frac{1}{1 - (x_{01} + x_{10} + x_{11})z^2}z^4 \right.$$
$$\left. \times [(x_{01} + x_{11})y_l + x_{10}][(x_{10} + x_{11})y_r + x_{01}] \right\} \frac{z^2}{1 - z^2}.$$

[1] J. I. Cirac, P. Zoller, H. J. Kimble, and H. Mabuchi, Phys. Rev. Lett. **78**, 3221 (1997).

[2] H. J. Kimble, Nature (London) **453**, 1023 (2008).

[3] C. H. Bennett, G. Brassard, C. Crépeau, R. Jozsa, A. Peres, and W. K. Wootters, Phys. Rev. Lett. **70**, 1895 (1993).

[4] N. Gisin, G. Ribordy, W. Tittel, and H. Zbinden, Rev. Mod. Phys. **74**, 145 (2002).

[5] S. Ritter, C. Nölleke, C. Hahn, A. Reiserer, A. Neuzner, M. Uphoff, M. Mücke, E. Figueroa, J. Bochmann, and G. Rempe, Nature (London) **484**, 195 (2012).

[6] B. Jungnitsch, T. Moroder, and O. Gühne, Phys. Rev. Lett. **106**, 190502 (2011).

[7] G. Paparo and M. Martin-Delgado, Scientific Reports **2**, 444 (2012).

[8] H. J. Briegel, W. Dür, J. I. Cirac, and P. Zoller, Phys. Rev. Lett. **81**, 5932 (1998).

[9] M. Zwerger, W. Dür, and H. J. Briegel, Phys. Rev. A **85**, 062326 (2012).

[10] A. Acín, J. I. Cirac, and M. Lewenstein, Nature Physics **3**, 256 (2007).

[11] M. Cuquet and J. Calsamiglia, Phys. Rev. Lett. **103**, 240503 (2009).

[12] S. Perseguers, Phys. Rev. A **81**, 012310 (2010).

[13] S. Perseguers, M. Lewenstein, A. Acín, and J. I. Cirac, Nature Physics **6**, 539 (2010).

[14] S. Broadfoot, U. Dorner, and D. Jaksch, Phys. Rev. A **81**, 042316 (2010).

[15] M. Cuquet and J. Calsamiglia, Phys. Rev. A **83**, 032319 (2011).

[16] S. Broadfoot, U. Dorner, and D. Jaksch, EPL (Europhysics Letters) **88**, 50002 (2009).

[17] M. Hein, J. Eisert, and H. J. Briegel, Phys. Rev. A **69**, 062311 (2004).

[18] D. Schlingemann and R. F. Werner, Phys. Rev. A **65**, 012308 (2001).

[19] W. Dür, J. Calsamiglia, and H.-J. Briegel, Phys. Rev. A **71**, 042336 (2005).

[20] O. Gühne, G. Tóth, P. Hyllus, and H. J. Briegel, Phys. Rev. Lett. **95**, 120405 (2005).

[21] H. J. Briegel, D. E. Browne, W. Dür, R. Raussendorf, and M. V. d. Nest, Nature Physics **5**, 19 (2009).

[22] M. Van den Nest, A. Miyake, W. Dür, and H. J. Briegel, Phys. Rev. Lett. **97**, 150504 (2006).

[23] C.-Y. Lu, X.-Q. Zhou, O. Gühne, W.-B. Gao, J. Zhang, Z.-S. Yuan, A. Goebel, T. Yang, and J.-W. Pan, Nature Physics **3**, 91 (2007).

[24] T. Monz, P. Schindler, J. T. Barreiro, M. Chwalla, D. Nigg, W. A. Coish, M. Harlander, W. Hänsel, M. Hennrich, and R. Blatt, Phys. Rev. Lett. **106**, 130506 (2011).

[25] R. Albert and A.-L. Barabási, Rev. Mod. Phys. **74**, 47 (2002).

[26] S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes, Rev. Mod. Phys. **80**, 1275 (2008).

[27] M. E. J. Newman, *Networks: An Introduction* (Oxford University Press, Oxford, 2010).

[28] H. Jeong, S. P. Mason, A.-L. Barabási, and Z. N. Oltvai, Nature (London) **411**, 41 (2001).

[29] A. Scala, L. A. N. Amaral, and M. Barthélémy, Europhysics Letters (EPL) **55**, 594 (2001).

[30] D. J. Watts and S. H. Strogatz, Nature (London) **393**, 440 (1998).

[31] L. A. N. Amaral, A. Scala, M. Barthélémy, and H. E. Stanley, Proc. Natl. Acad. Sci. USA **97**, 11149 (2000).

[32] R. Pastor-Satorras, A. Vázquez, and A. Vespignani, Phys. Rev. Lett. **87**, 258701 (2001).

[33] S.-H. Yook, H. Jeong, and A.-L. Barabási, Proc. Natl. Acad. Sci. USA **99**, 13382 (2002).

[34] M. E. J. Newman, S. H. Strogatz, and D. J. Watts, Phys. Rev. E **64**, 026118 (2001).

[35] D. Wiersma, Science **327**, 1333 (2010).

[36] O. Mülken and A. Blumen, Physics Reports **502**, 37 (2011).

[37] C. H. Bennett, G. Brassard, S. Popescu, B. Schumacher, J. A. Smolin, and W. K. Wootters, Phys. Rev. Lett. **76**, 722 (1996).

[38] D. Deutsch, A. K. Ekert, R. Jozsa, C. Macchiavello, S. Popescu, and A. Sanpera, Phys. Rev. Lett. **77**, 2818 (1996).

[39] W. Dür, H. Aschauer, and H. J. Briegel, Phys. Rev. Lett. **91**, 107903 (2003).

[40] H. Aschauer, W. Dür, and H. J. Briegel, Phys. Rev. A **71**, 012319 (2005).

[41] C. Kruszynska, A. Miyake, H. J. Briegel, and W. Dür, Phys. Rev. A **74**, 052316 (2006).

[42] S. Glancy, E. Knill, and H. M. Vasconcelos, Phys. Rev. A **74**, 032319 (2006).

[43] A. Kay and J. K. Pachos, Phys. Rev. A **75**, 062307 (2007).

[44] W. Dür and H. J. Briegel, Rep. Prog. Phys. **70**, 1381 (2007).

[45] K. Goyal, A. McCauley, and R. Raussendorf, Phys. Rev. A **74**, 032318 (2006).

[46] E. T. Campbell, Phys. Rev. A **76**, 040302(R) (2007).

[47] K. Kieling, D. Gross, and J. Eisert, New J. Phys. **9**, 200 (2007).

[48] P. P. Rohde and S. D. Barrett, New J. Phys. **9**, 198 (2007).

[49] E. T. Campbell, J. Fitzsimons, S. C. Benjamin, and P. Kok, Phys. Rev. A **75**, 042303 (2007).

[50] E. T. Campbell, J. Fitzsimons, S. C. Benjamin, and P. Kok, New J. Phys. **9**, 196 (2007).

[51] Y. Matsuzaki, S. C. Benjamin, and J. Fitzsimons, Phys. Rev. Lett. **104**, 050501 (2010).

[52] M. Van den Nest, J. Dehaene, and B. De Moor, Phys. Rev. A **69**, 022316 (2004).

[53] J. Joo and D. L Feder, New J. Phys. **13**, 063025 (2011).

[54] D. J. Amit, *Modeling Brain Function* (Cambridge University Press, Cambridge, 1989).

[55] M. Mézard, G. Parisi, and M. A. Virasoro, *Spin Glass Theory and Beyond* (World Scientific Publishing, Singapore, 1987).

[56] E. N. Gilbert, The Annals of Mathematical Statistics **30**, 1141 (1959).

[57] P. Erdős and A. Rényi, Publicationes Mathematicae (Debrecen) **6**, 290 (1959).

[58] P. Erdős and A. Rényi, Publ. Math. Inst. Hung. Acad. Sci **5**, 17 (1960).